Package ‘whitebox’
March 21, 2022

Type Package
Title 'WhiteboxTools' R Frontend
Version 2.1.2
Description An R frontend for the 'WhiteboxTools' library, which is an advanced geospatial data analysis platform developed by Prof. John Lindsay at the University of Guelph's Geomorphometry and Hydrogeomatics Research Group. 'WhiteboxTools' can be used to perform common geographical information systems (GIS) analysis operations, such as cost-distance analysis, distance buffering, and raster reclassification. Remote sensing and image processing tasks include image enhancement (e.g. panchromatic sharpening, contrast adjustments), image mosaicing, numerous filtering operations, simple classification (k-means), and common image transformations. 'WhiteboxTools' also contains advanced tooling for spatial hydrological analysis (e.g. flow-accumulation, watershed delineation, stream network analysis, sink removal), terrain analysis (e.g. common terrain indices such as slope, curvatures, wetness index, hillshading; hypsometric analysis; multi-scale topographic position analysis), and LiDAR data processing. Suggested citation: Lindsay (2016) <doi:10.1016/j.cageo.2016.07.003>.
Maintainer Andrew Brown <brown.andrewg@gmail.com>
License MIT + file LICENSE
SystemRequirements WhiteboxTools
  (https://github.com/jblindsay/whitebox-tools/releases/latest)
Encoding UTF-8
RoxygenNote 7.1.2
URL https://github.com/giswqs/whiteboxR
BugReports https://github.com/giswqs/whiteboxR/issues
Suggests knitr, rmarkdown, testthat, raster, rgdal
VignetteBuilder knitr
Depends R (>= 2.10)
NeedsCompilation no
Author Qiusheng Wu [aut],
  Andrew Brown [ctb, cre]
Repository CRAN
Date/Publication 2022-03-21 07:50:02 UTC
R topics documented:

- check_whitebox_binary .................................................. 13
- sample_dem_data .......................................................... 13
- wbttoolparameters ......................................................... 14
- wbttools ................................................................. 15
- wbt_absolute_value ...................................................... 15
- wbt_accumulation_curvature ............................................ 16
- wbt_adaptive_filter ....................................................... 17
- wbt_add ................................................................. 18
- wbt_add_point_coordinates_to_table ................................ 19
- wbt_aggregate_raster ..................................................... 19
- wbt_and ................................................................. 20
- wbt_anova ............................................................... 21
- wbt_arcosh ............................................................. 22
- wbt_arc_cos ............................................................ 23
- wbt_arc_sin ............................................................. 24
- wbt_arc_tan ............................................................. 24
- wbt_arsinh .............................................................. 25
- wbt_artanh ............................................................. 26
- wbt_ascii_to_las ......................................................... 27
- wbt_aspect ............................................................... 28
- wbt_assess_route .......................................................... 29
- wbt_atan2 ............................................................... 30
- wbt_attribute_correlation .............................................. 31
- wbt_attribute_correlation_neighbourhood_analysis .............. 31
- wbt_attribute_histogram ................................................ 32
- wbt_attribute_scattergram .............................................. 33
- wbt_average_flowpath_slope .......................................... 34
- wbt_average_normal_vector-angular_deviation .................... 35
- wbt_average_overlay .................................................... 36
- wbt_average_upslope_flowpath_length ............................... 37
- wbt_balance_contrast_enhancement .................................. 37
- wbt_basins ............................................................. 38
- wbt_bilateral_filter ..................................................... 39
- wbt_block_maximum_gridding .......................................... 40
- wbt_block_minimum_gridding .......................................... 41
- wbt_boundary_shape_complexity ...................................... 42
- wbt_breach_depressions ................................................ 43
- wbt_breach_depressions_least_cost ................................ 44
- wbt_breach_single_cell_pits .......................................... 45
- wbt_buffer_raster ....................................................... 45
- wbt_burn_streams_at_roads .......................................... 46
- wbt_canny_edge_detection ............................................ 47
- wbt_ceil ............................................................... 48
- wbt_centroid ........................................................... 49
- wbt_centroid_vector ..................................................... 50
- wbt_change_vector_analysis .......................................... 51
<table>
<thead>
<tr>
<th>Function Name</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>wbt_circular_variance_of_aspect</td>
<td>52</td>
</tr>
<tr>
<td>wbt_classify_buildings_in_lidar</td>
<td>53</td>
</tr>
<tr>
<td>wbt_classify_overlap_points</td>
<td>54</td>
</tr>
<tr>
<td>wbt_clean_vector</td>
<td>55</td>
</tr>
<tr>
<td>wbt_clip</td>
<td>55</td>
</tr>
<tr>
<td>wbt_clip_lidar_to_polygon</td>
<td>56</td>
</tr>
<tr>
<td>wbt_clip_raster_to_polygon</td>
<td>57</td>
</tr>
<tr>
<td>wbt_closing</td>
<td>58</td>
</tr>
<tr>
<td>wbt_clump</td>
<td>59</td>
</tr>
<tr>
<td>wbt_compactness_ratio</td>
<td>60</td>
</tr>
<tr>
<td>wbt_conditional_evaluation</td>
<td>60</td>
</tr>
<tr>
<td>wbt_conservative_smoothing_filter</td>
<td>61</td>
</tr>
<tr>
<td>wbt_construct_vector_tin</td>
<td>62</td>
</tr>
<tr>
<td>wbt_contours_from_points</td>
<td>63</td>
</tr>
<tr>
<td>wbt_contours_from_raster</td>
<td>64</td>
</tr>
<tr>
<td>wbt_convert_nodata_to_zero</td>
<td>65</td>
</tr>
<tr>
<td>wbt_convert_raster_format</td>
<td>66</td>
</tr>
<tr>
<td>wbt_corner_detection</td>
<td>67</td>
</tr>
<tr>
<td>wbt_correct_vignetting</td>
<td>68</td>
</tr>
<tr>
<td>wbt_cos</td>
<td>69</td>
</tr>
<tr>
<td>wbt_cosh</td>
<td>69</td>
</tr>
<tr>
<td>wbt_cost_allocation</td>
<td>70</td>
</tr>
<tr>
<td>wbt_cost_distance</td>
<td>71</td>
</tr>
<tr>
<td>wbt_cost_pathway</td>
<td>72</td>
</tr>
<tr>
<td>wbt_count_if</td>
<td>73</td>
</tr>
<tr>
<td>wbt_create_colour_composite</td>
<td>74</td>
</tr>
<tr>
<td>wbt_create_hexagonal_vector_grid</td>
<td>75</td>
</tr>
<tr>
<td>wbt_create_plane</td>
<td>76</td>
</tr>
<tr>
<td>wbt_create_rectangular_vector_grid</td>
<td>77</td>
</tr>
<tr>
<td>wbt_crispness_index</td>
<td>78</td>
</tr>
<tr>
<td>wbt_cross_tabulation</td>
<td>78</td>
</tr>
<tr>
<td>wbt_csv_points_to_vector</td>
<td>79</td>
</tr>
<tr>
<td>wbt_cumulative_distribution</td>
<td>80</td>
</tr>
<tr>
<td>wbt_curvedness</td>
<td>81</td>
</tr>
<tr>
<td>wbt_d8_flow_accumulation</td>
<td>82</td>
</tr>
<tr>
<td>wbt_d8_mass_flux</td>
<td>83</td>
</tr>
<tr>
<td>wbt_d8_pointer</td>
<td>84</td>
</tr>
<tr>
<td>wbt_dbscan</td>
<td>85</td>
</tr>
<tr>
<td>wbt_decrement</td>
<td>86</td>
</tr>
<tr>
<td>wbt_depth_in_sink</td>
<td>86</td>
</tr>
<tr>
<td>wbt_dev_from_mean_elev</td>
<td>87</td>
</tr>
<tr>
<td>wbt_difference</td>
<td>88</td>
</tr>
<tr>
<td>wbt_difference_curvature</td>
<td>89</td>
</tr>
<tr>
<td>wbt_diff_from_mean_elev</td>
<td>90</td>
</tr>
<tr>
<td>wbt_diff_of_gaussian_filter</td>
<td>91</td>
</tr>
<tr>
<td>wbt_directional_relief</td>
<td>92</td>
</tr>
<tr>
<td>wbt_direct_decorrelation_stretch</td>
<td>93</td>
</tr>
<tr>
<td>wbt_dissolve</td>
<td>94</td>
</tr>
</tbody>
</table>
R topics documented:

<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>wbt_distance_to_outlet</td>
<td>95</td>
</tr>
<tr>
<td>wbt_diversity_filter</td>
<td>96</td>
</tr>
<tr>
<td>wbt_divide</td>
<td>97</td>
</tr>
<tr>
<td>wbt_downslope_distance_to_stream</td>
<td>98</td>
</tr>
<tr>
<td>wbt_downslope_flowpath_length</td>
<td>99</td>
</tr>
<tr>
<td>wbt_downslope_index</td>
<td>100</td>
</tr>
<tr>
<td>wbt_d_inf_flow_accumulation</td>
<td>101</td>
</tr>
<tr>
<td>wbt_d_inf_mass_flux</td>
<td>102</td>
</tr>
<tr>
<td>wbt_d_inf_pointer</td>
<td>103</td>
</tr>
<tr>
<td>wbt_edge_contamination</td>
<td>103</td>
</tr>
<tr>
<td>wbt_edge_density</td>
<td>104</td>
</tr>
<tr>
<td>wbt_edge_preserving_mean_filter</td>
<td>105</td>
</tr>
<tr>
<td>wbt_edge_proportion</td>
<td>106</td>
</tr>
<tr>
<td>wbt_elevation_above_stream</td>
<td>107</td>
</tr>
<tr>
<td>wbt_elevation_above_stream_euclidean</td>
<td>108</td>
</tr>
<tr>
<td>wbt_elev_above_pit</td>
<td>109</td>
</tr>
<tr>
<td>wbt_elev_percentile</td>
<td>109</td>
</tr>
<tr>
<td>wbt_elev_relative_to_min_max</td>
<td>110</td>
</tr>
<tr>
<td>wbt_elev_relative_to_watershed_min_max</td>
<td>111</td>
</tr>
<tr>
<td>wbt_eliminate_coincident_points</td>
<td>112</td>
</tr>
<tr>
<td>wbt_elongation_ratio</td>
<td>113</td>
</tr>
<tr>
<td>wbt_embankment_mapping</td>
<td>114</td>
</tr>
<tr>
<td>wbt_emboss_filter</td>
<td>115</td>
</tr>
<tr>
<td>wbt_equal_to</td>
<td>116</td>
</tr>
<tr>
<td>wbt_erase</td>
<td>117</td>
</tr>
<tr>
<td>wbt_erase_polygon_from_lidar</td>
<td>118</td>
</tr>
<tr>
<td>wbt_erase_polygon_from_raster</td>
<td>119</td>
</tr>
<tr>
<td>wbt_euclidean_allocation</td>
<td>120</td>
</tr>
<tr>
<td>wbt_euclidean_distance</td>
<td>120</td>
</tr>
<tr>
<td>wbt_evaluate_training_sites</td>
<td>121</td>
</tr>
<tr>
<td>wbt_exp</td>
<td>122</td>
</tr>
<tr>
<td>wbt_exp2</td>
<td>123</td>
</tr>
<tr>
<td>wbt_export_table_to_csv</td>
<td>124</td>
</tr>
<tr>
<td>wbt_exposure_towards_wind_flux</td>
<td>125</td>
</tr>
<tr>
<td>wbt_extend_vector_lines</td>
<td>126</td>
</tr>
<tr>
<td>wbt_extract_nodes</td>
<td>127</td>
</tr>
<tr>
<td>wbt_extract_raster_values_at_points</td>
<td>127</td>
</tr>
<tr>
<td>wbt_extract_streams</td>
<td>128</td>
</tr>
<tr>
<td>wbt_extract_valleys</td>
<td>129</td>
</tr>
<tr>
<td>wbt_farthest_channel_head</td>
<td>130</td>
</tr>
<tr>
<td>wbt_fast_almost_gaussian_filter</td>
<td>131</td>
</tr>
<tr>
<td>wbt_fd8_flow_accumulation</td>
<td>132</td>
</tr>
<tr>
<td>wbt_fd8_pointer</td>
<td>133</td>
</tr>
<tr>
<td>wbt_feature_preserving_smoothing</td>
<td>134</td>
</tr>
<tr>
<td>wbt_fetch_analysis</td>
<td>135</td>
</tr>
<tr>
<td>wbt_fill_burn</td>
<td>136</td>
</tr>
<tr>
<td>wbt_fill_depressions</td>
<td>137</td>
</tr>
<tr>
<td>wbt_fill_depressions_planchon_and_darboux</td>
<td>138</td>
</tr>
<tr>
<td>R topics documented:</td>
<td></td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>-----</td>
</tr>
<tr>
<td>wbt_fill_depressions_wang_and_liu</td>
<td>139</td>
</tr>
<tr>
<td>wbt_fill_missing_data</td>
<td>140</td>
</tr>
<tr>
<td>wbt_fill_single_cell_pits</td>
<td>141</td>
</tr>
<tr>
<td>wbt_filter_lidar_classes</td>
<td>141</td>
</tr>
<tr>
<td>wbt_filter_lidar_scan_angles</td>
<td>142</td>
</tr>
<tr>
<td>wbt_filter_raster_features_by_area</td>
<td>143</td>
</tr>
<tr>
<td>wbt_find_flightline_edge_points</td>
<td>144</td>
</tr>
<tr>
<td>wbt_find_lowest_or_highest_points</td>
<td>145</td>
</tr>
<tr>
<td>wbt_find_main_stem</td>
<td>146</td>
</tr>
<tr>
<td>wbt_find_no_flow_cells</td>
<td>147</td>
</tr>
<tr>
<td>wbt_find_parallel_flow</td>
<td>147</td>
</tr>
<tr>
<td>wbt_find_patch_or_class_edge_cells</td>
<td>148</td>
</tr>
<tr>
<td>wbt_find_ridges</td>
<td>149</td>
</tr>
<tr>
<td>wbt_fix_dangling_arcs</td>
<td>150</td>
</tr>
<tr>
<td>wbt_flatten_lakes</td>
<td>151</td>
</tr>
<tr>
<td>wbt_flightline_overlap</td>
<td>152</td>
</tr>
<tr>
<td>wbt_flip_image</td>
<td>153</td>
</tr>
<tr>
<td>wbt_flood_order</td>
<td>154</td>
</tr>
<tr>
<td>wbt_floor</td>
<td>154</td>
</tr>
<tr>
<td>wbt_flow_accumulation_full_workflow</td>
<td>155</td>
</tr>
<tr>
<td>wbt_flow_length_diff</td>
<td>156</td>
</tr>
<tr>
<td>wbt_gamma_correction</td>
<td>157</td>
</tr>
<tr>
<td>wbt_gaussian_contrast_stretch</td>
<td>158</td>
</tr>
<tr>
<td>wbt_gaussian_curvature</td>
<td>159</td>
</tr>
<tr>
<td>wbt_gaussian_filter</td>
<td>160</td>
</tr>
<tr>
<td>wbt_gaussian_scale_space</td>
<td>161</td>
</tr>
<tr>
<td>wbt_generalize_classified_raster</td>
<td>162</td>
</tr>
<tr>
<td>wbt_generalize_with_similarity</td>
<td>163</td>
</tr>
<tr>
<td>wbt_generating_function</td>
<td>164</td>
</tr>
<tr>
<td>wbt_geomorphons</td>
<td>165</td>
</tr>
<tr>
<td>wbt_greater_than</td>
<td>166</td>
</tr>
<tr>
<td>wbt_hack_stream_order</td>
<td>167</td>
</tr>
<tr>
<td>wbt_height_above_ground</td>
<td>168</td>
</tr>
<tr>
<td>wbt_help</td>
<td>168</td>
</tr>
<tr>
<td>wbt_highest_position</td>
<td>169</td>
</tr>
<tr>
<td>wbt_high_pass_filter</td>
<td>170</td>
</tr>
<tr>
<td>wbt_high_pass_median_filter</td>
<td>171</td>
</tr>
<tr>
<td>wbt_hillshade</td>
<td>172</td>
</tr>
<tr>
<td>wbt_hillslopes</td>
<td>173</td>
</tr>
<tr>
<td>wbt_histogram_equalization</td>
<td>174</td>
</tr>
<tr>
<td>wbt_histogram_matching</td>
<td>175</td>
</tr>
<tr>
<td>wbt_histogram_matching_two_images</td>
<td>176</td>
</tr>
<tr>
<td>wbt_hole_proportion</td>
<td>177</td>
</tr>
<tr>
<td>wbt_horizontal_excess_curvature</td>
<td>177</td>
</tr>
<tr>
<td>wbt_horizon_angle</td>
<td>178</td>
</tr>
<tr>
<td>wbt_horton_stream_order</td>
<td>179</td>
</tr>
<tr>
<td>wbt_hydrologic_connectivity</td>
<td>180</td>
</tr>
<tr>
<td>wbt_hypsometrically_tinted_hillshade</td>
<td>181</td>
</tr>
<tr>
<td>Topic</td>
<td>Page</td>
</tr>
<tr>
<td>---------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>wbt_hypsometric_analysis</td>
<td>182</td>
</tr>
<tr>
<td>wbt_idw_interpolation</td>
<td>183</td>
</tr>
<tr>
<td>wbt_ihs_to_rgb</td>
<td>184</td>
</tr>
<tr>
<td>wbt_image_autocorrelation</td>
<td>185</td>
</tr>
<tr>
<td>wbt_image_correlation</td>
<td>186</td>
</tr>
<tr>
<td>wbt_image_correlation_neighbourhood_analysis</td>
<td>187</td>
</tr>
<tr>
<td>wbt_image_regression</td>
<td>188</td>
</tr>
<tr>
<td>wbt_image_segmentation</td>
<td>189</td>
</tr>
<tr>
<td>wbt_image_slider</td>
<td>190</td>
</tr>
<tr>
<td>wbt_image_stack_profile</td>
<td>191</td>
</tr>
<tr>
<td>wbt_impoundment_size_index</td>
<td>192</td>
</tr>
<tr>
<td>wbt_increment</td>
<td>193</td>
</tr>
<tr>
<td>wbt_init</td>
<td>193</td>
</tr>
<tr>
<td>wbt_insert_dams</td>
<td>197</td>
</tr>
<tr>
<td>wbt_install</td>
<td>198</td>
</tr>
<tr>
<td>wbt_integer_division</td>
<td>198</td>
</tr>
<tr>
<td>wbt_integral_image</td>
<td>199</td>
</tr>
<tr>
<td>wbt_intersect</td>
<td>200</td>
</tr>
<tr>
<td>wbt_inverse_principal_component_analysis</td>
<td>201</td>
</tr>
<tr>
<td>wbt_in_place_add</td>
<td>202</td>
</tr>
<tr>
<td>wbt_in_place_divide</td>
<td>202</td>
</tr>
<tr>
<td>wbt_in_place_multiply</td>
<td>203</td>
</tr>
<tr>
<td>wbt_in_place_subtract</td>
<td>204</td>
</tr>
<tr>
<td>wbt_isobasins</td>
<td>205</td>
</tr>
<tr>
<td>wbt_is_no_data</td>
<td>206</td>
</tr>
<tr>
<td>wbt_jenson_snap_pour_points</td>
<td>206</td>
</tr>
<tr>
<td>wbt_join_tables</td>
<td>207</td>
</tr>
<tr>
<td>wbt_kappa_index</td>
<td>208</td>
</tr>
<tr>
<td>wbt_knn_classification</td>
<td>209</td>
</tr>
<tr>
<td>wbt_knn_regression</td>
<td>210</td>
</tr>
<tr>
<td>wbt_ks_test_for_normality</td>
<td>211</td>
</tr>
<tr>
<td>wbt_k_means_clustering</td>
<td>212</td>
</tr>
<tr>
<td>wbt_k_nearest_mean_filter</td>
<td>213</td>
</tr>
<tr>
<td>wbt_laplacian_filter</td>
<td>214</td>
</tr>
<tr>
<td>wbt_laplacian_of_gaussian_filter</td>
<td>215</td>
</tr>
<tr>
<td>wbt_las_to_ascii</td>
<td>216</td>
</tr>
<tr>
<td>wbt_las_to_laz</td>
<td>217</td>
</tr>
<tr>
<td>wbt_las_to_multipoint_shapefile</td>
<td>217</td>
</tr>
<tr>
<td>wbt_las_to_shapefile</td>
<td>218</td>
</tr>
<tr>
<td>wbt_las_to_zidar</td>
<td>219</td>
</tr>
<tr>
<td>wbt_layer_footprint</td>
<td>220</td>
</tr>
<tr>
<td>wbt_laz_to_las</td>
<td>221</td>
</tr>
<tr>
<td>wbt_lee_sigma_filter</td>
<td>221</td>
</tr>
<tr>
<td>wbt_length_of_upstream_channels</td>
<td>222</td>
</tr>
<tr>
<td>wbt_less_than</td>
<td>223</td>
</tr>
<tr>
<td>wbt_license</td>
<td>224</td>
</tr>
<tr>
<td>wbt_lidar_block_maximum</td>
<td>225</td>
</tr>
<tr>
<td>wbt_lidar_block_minimum</td>
<td>226</td>
</tr>
</tbody>
</table>
R topics documented:

- `wbt_lidar_classify_subset` .................................................. 227
- `wbt_lidar_colourize` ......................................................... 228
- `wbt_lidar_contour` ............................................................ 229
- `wbt_lidar.digital_surface_model` .......................................... 230
- `wbt_lidar.elevation_slice` .................................................. 231
- `wbt_lidar.ground_point_filter` ............................................. 232
- `wbt_lidar.hex_binning` ..................................................... 233
- `wbt_lidar.hillshade` ......................................................... 234
- `wbt_lidar.histogram` .......................................................... 235
- `wbt_lidar.idw_interpolation` ............................................... 236
- `wbt_lidar.info` ............................................................... 237
- `wbt_lidar.join` ............................................................... 238
- `wbt_lidar.kappa_index` ...................................................... 239
- `wbt_lidar.nearest_neighbour_gridding` .................................. 240
- `wbt_lidar.point_density` ................................................... 241
- `wbt_lidar.point_return_analysis` ......................................... 242
- `wbt_lidar.point_stats` ........................................................ 243
- `wbt_lidar.ransac_planes` ................................................... 244
- `wbt_lidar.rbf_interpolation` ............................................... 245
- `wbt_lidar.remove_duplicates` ............................................. 246
- `wbt_lidar.remove_outliers` ................................................ 247
- `wbt_lidar.rooftop_analysis` ............................................... 248
- `wbt_lidar.segmentation` .................................................... 249
- `wbt_lidar.segmentation_based_filter` .................................... 250
- `wbt_lidar.shift` ............................................................. 251
- `wbt_lidar.sibson_interpolation` .......................................... 252
- `wbt_lidar.sort_by_time` .................................................... 253
- `wbt_lidar.thin` .............................................................. 254
- `wbt_lidar.thin.high_density` ............................................. 255
- `wbt_lidar.tile` ............................................................... 256
- `wbt_lidar.tile_footprint` ................................................... 257
- `wbt_lidar.tin_gridding` .................................................... 258
- `wbt_lidar.tophat_transform` ............................................... 259
- `wbt_linearity_index` ........................................................ 260
- `wbt_local.hypsometric_analysis` ......................................... 261
- `wbt_local.quadratic_regression` .......................................... 262
- `wbt_log10` ................................................................. 263
- `wbt_log2` ................................................................. 264
- `wbt_logistic.regression` ................................................... 265
- `wbt.longest.flowpath` ...................................................... 266
- `wbt.local.hypsometric_analysis` .......................................... 267
- `wbt.longest.flowpath` ...................................................... 268
- `wbt_log10` ................................................................. 269
- `wbt_log2` ................................................................. 269
- `wbt_logistic.regression` ................................................... 270
- `wbt.longest.flowpath` ...................................................... 271
- `wbt.long.profile` .......................................................... 272
<table>
<thead>
<tr>
<th>Function Name</th>
<th>Page</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>wbt_long_profile_from_points</td>
<td>273</td>
<td></td>
</tr>
<tr>
<td>wbt_lowest_position</td>
<td>274</td>
<td></td>
</tr>
<tr>
<td>wbt_low_points_on_headwater_divides</td>
<td>275</td>
<td></td>
</tr>
<tr>
<td>wbt_majority_filter</td>
<td>276</td>
<td></td>
</tr>
<tr>
<td>wbt_map_off_terrain_objects</td>
<td>277</td>
<td></td>
</tr>
<tr>
<td>wbt_max</td>
<td>278</td>
<td></td>
</tr>
<tr>
<td>wbt_maximal_curvature</td>
<td>279</td>
<td></td>
</tr>
<tr>
<td>wbt_maximum_filter</td>
<td>280</td>
<td></td>
</tr>
<tr>
<td>wbt_max_absolute_overlay</td>
<td>281</td>
<td></td>
</tr>
<tr>
<td>wbt_max_anisotropy_dev</td>
<td>281</td>
<td></td>
</tr>
<tr>
<td>wbt_max_anisotropy_dev_signature</td>
<td>282</td>
<td></td>
</tr>
<tr>
<td>wbt_max_branch_length</td>
<td>283</td>
<td></td>
</tr>
<tr>
<td>wbt_max_difference_from_mean</td>
<td>284</td>
<td></td>
</tr>
<tr>
<td>wbt_max_downslope_elev_change</td>
<td>285</td>
<td></td>
</tr>
<tr>
<td>wbt_max_elevation_dev</td>
<td>286</td>
<td></td>
</tr>
<tr>
<td>wbt_max_elev_dev_signature</td>
<td>287</td>
<td></td>
</tr>
<tr>
<td>wbt_max_overlay</td>
<td>288</td>
<td></td>
</tr>
<tr>
<td>wbt_max_upslope_elev_change</td>
<td>289</td>
<td></td>
</tr>
<tr>
<td>wbt_max_upslope_flowpath_length</td>
<td>289</td>
<td></td>
</tr>
<tr>
<td>wbt_md_inf_flow_accumulation</td>
<td>290</td>
<td></td>
</tr>
<tr>
<td>wbt_mean_curvature</td>
<td>291</td>
<td></td>
</tr>
<tr>
<td>wbt_mean_filter</td>
<td>292</td>
<td></td>
</tr>
<tr>
<td>wbt_median_filter</td>
<td>293</td>
<td></td>
</tr>
<tr>
<td>wbt_medoid</td>
<td>294</td>
<td></td>
</tr>
<tr>
<td>wbt_merge_line_segments</td>
<td>295</td>
<td></td>
</tr>
<tr>
<td>wbt_merge_table_with_csv</td>
<td>296</td>
<td></td>
</tr>
<tr>
<td>wbt_merge_vectors</td>
<td>297</td>
<td></td>
</tr>
<tr>
<td>wbt_min</td>
<td>297</td>
<td></td>
</tr>
<tr>
<td>wbt_minimal_curvature</td>
<td>298</td>
<td></td>
</tr>
<tr>
<td>wbt_minimum_bounding_box</td>
<td>299</td>
<td></td>
</tr>
<tr>
<td>wbt_minimum_bounding_circle</td>
<td>300</td>
<td></td>
</tr>
<tr>
<td>wbt_minimum_bounding_envelope</td>
<td>301</td>
<td></td>
</tr>
<tr>
<td>wbt_minimum_convex_hull</td>
<td>302</td>
<td></td>
</tr>
<tr>
<td>wbt_minimum_filter</td>
<td>303</td>
<td></td>
</tr>
<tr>
<td>wbt_min_absolute_overlay</td>
<td>304</td>
<td></td>
</tr>
<tr>
<td>wbt_min_dist_classification</td>
<td>304</td>
<td></td>
</tr>
<tr>
<td>wbt_min_downslope_elev_change</td>
<td>305</td>
<td></td>
</tr>
<tr>
<td>wbt_min_max_contrast_stretch</td>
<td>306</td>
<td></td>
</tr>
<tr>
<td>wbt_min_overlay</td>
<td>307</td>
<td></td>
</tr>
<tr>
<td>wbt_modified_k_means_clustering</td>
<td>308</td>
<td></td>
</tr>
<tr>
<td>wbt_modify_no_data_value</td>
<td>309</td>
<td></td>
</tr>
<tr>
<td>wbt_modulo</td>
<td>310</td>
<td></td>
</tr>
<tr>
<td>wbt_mosaic</td>
<td>311</td>
<td></td>
</tr>
<tr>
<td>wbt_mosaic_with_feathering</td>
<td>312</td>
<td></td>
</tr>
<tr>
<td>wbt_multidirectional_hillshade</td>
<td>313</td>
<td></td>
</tr>
<tr>
<td>wbt_multiply</td>
<td>314</td>
<td></td>
</tr>
<tr>
<td>wbt_multiscale_elevation_percentile</td>
<td>315</td>
<td></td>
</tr>
<tr>
<td>wbt_multiscale_roughness</td>
<td>316</td>
<td></td>
</tr>
</tbody>
</table>
R topics documented:

- wbt_multiscale_roughness_signature ........................................ 317
- wbt_multiscale_std_dev_normals ............................................. 318
- wbt_multiscale_std_dev_normals_signature ............................... 319
- wbt_multiscale_topographic_position_image .............................. 320
- wbt_multi_part_to_single_part ............................................. 321
- wbt_narrowness_index ..................................................... 322
- wbt_natural_neighbour_interpolation .................................... 322
- wbt_nearest_neighbour_gridding ......................................... 323
- wbt_negate ................................................................. 324
- wbt_new_raster_from_base .................................................. 325
- wbt_normalized_difference_index .......................................... 326
- wbt_normal_vectors ....................................................... 327
- wbt_not ................................................................. 328
- wbt_not_equal_to ......................................................... 329
- wbt_num_downslope_neighbours ........................................... 330
- wbt_num_inflowing_neighbours ............................................ 330
- wbt_num_upslope_neighbours .............................................. 331
- wbt_olympic_filter ....................................................... 332
- wbt_opening ............................................................ 333
- wbt_openness ............................................................ 334
- wbt_or ................................................................. 335
- wbt_paired_sample_t_test .................................................. 336
- wbt_panchromatic_sharpening ............................................. 337
- wbt_parallelepiped_classification ....................................... 338
- wbt_patch_orientation ..................................................... 339
- wbt_pennock_landform_class .............................................. 339
- wbt_percentage_contrast_stretch ......................................... 340
- wbt_percentile_filter ..................................................... 341
- wbt_percent_elev_range ................................................... 342
- wbt_percent_equal_to ..................................................... 343
- wbt_percent_greater_than ............................................... 344
- wbt_percent_less_than .................................................... 345
- wbt_perimeter_area_ratio ................................................. 346
- wbt_phi_coefficient ....................................................... 347
- wbt_pick_from_list ....................................................... 348
- wbt_plan_curvature ....................................................... 349
- wbt_polygonize .......................................................... 350
- wbt_polygons_to_lines ..................................................... 350
- wbt_polygon_area ........................................................ 351
- wbt_polygon_long_axis .................................................... 352
- wbt_polygon_perimeter .................................................... 353
- wbt_polygon_short_axis ................................................... 353
- wbt_power ............................................................... 354
- wbt_prewitt_filter ....................................................... 355
- wbt_principal_component_analysis ..................................... 356
- wbt_print_geo_tiff_tags ................................................... 357
- wbt_profile ............................................................ 357
- wbt_profile_curvature ..................................................... 358
<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>wbt_qin_flow_accumulation</td>
<td>359</td>
</tr>
<tr>
<td>wbt_quantiles</td>
<td>360</td>
</tr>
<tr>
<td>wbt_quin_flow_accumulation</td>
<td>361</td>
</tr>
<tr>
<td>wbt_radial_basis_function_interpolation</td>
<td>362</td>
</tr>
<tr>
<td>wbt_radius_of_gyration</td>
<td>363</td>
</tr>
<tr>
<td>wbt_raise_walls</td>
<td>364</td>
</tr>
<tr>
<td>wbt_random_field</td>
<td>365</td>
</tr>
<tr>
<td>wbt_random_forest_classification</td>
<td>366</td>
</tr>
<tr>
<td>wbt_random_forest_regression</td>
<td>367</td>
</tr>
<tr>
<td>wbt_random_sample</td>
<td>368</td>
</tr>
<tr>
<td>wbt_range_filter</td>
<td>369</td>
</tr>
<tr>
<td>wbt_rasterize_streams</td>
<td>370</td>
</tr>
<tr>
<td>wbt_raster_area</td>
<td>371</td>
</tr>
<tr>
<td>wbt_raster_calculator</td>
<td>372</td>
</tr>
<tr>
<td>wbt_raster_cell_assignment</td>
<td>373</td>
</tr>
<tr>
<td>wbt_raster_histogram</td>
<td>374</td>
</tr>
<tr>
<td>wbt_raster_perimeter</td>
<td>375</td>
</tr>
<tr>
<td>wbt_raster_streams_to_vector</td>
<td>376</td>
</tr>
<tr>
<td>wbt_raster_summary_stats</td>
<td>377</td>
</tr>
<tr>
<td>wbt_raster_to_vector_lines</td>
<td>377</td>
</tr>
<tr>
<td>wbt_raster_to_vector_points</td>
<td>378</td>
</tr>
<tr>
<td>wbt_raster_to_vector_polygons</td>
<td>379</td>
</tr>
<tr>
<td>wbt_reciprocal</td>
<td>380</td>
</tr>
<tr>
<td>wbt_reclass</td>
<td>381</td>
</tr>
<tr>
<td>wbt_reclass_equal_interval</td>
<td>382</td>
</tr>
<tr>
<td>wbt_reclass_from_file</td>
<td>383</td>
</tr>
<tr>
<td>wbt_reconcile_multiple_headers</td>
<td>384</td>
</tr>
<tr>
<td>wbt_recreate_pass_lines</td>
<td>385</td>
</tr>
<tr>
<td>wbt_reinitialize_attribute_table</td>
<td>386</td>
</tr>
<tr>
<td>wbt_related_circumscribing_circle</td>
<td>387</td>
</tr>
<tr>
<td>wbt_relative_aspect</td>
<td>387</td>
</tr>
<tr>
<td>wbt_relative_topographic_position</td>
<td>388</td>
</tr>
<tr>
<td>wbt_remove_field_edge_points</td>
<td>389</td>
</tr>
<tr>
<td>wbt_remove_off_terrain_objects</td>
<td>390</td>
</tr>
<tr>
<td>wbt_remove_polygon_holes</td>
<td>391</td>
</tr>
<tr>
<td>wbt_remove_short_streams</td>
<td>392</td>
</tr>
<tr>
<td>wbt_remove_spurs</td>
<td>393</td>
</tr>
<tr>
<td>wbt_repair_stream_vector_topology</td>
<td>394</td>
</tr>
<tr>
<td>wbt_resample</td>
<td>395</td>
</tr>
<tr>
<td>wbt_rescale_value_range</td>
<td>396</td>
</tr>
<tr>
<td>wbt_rgb_to_ihs</td>
<td>397</td>
</tr>
<tr>
<td>wbt_rho8_flow_accumulation</td>
<td>398</td>
</tr>
<tr>
<td>wbt_rho8_pointer</td>
<td>399</td>
</tr>
<tr>
<td>wbt_ring_curvature</td>
<td>400</td>
</tr>
<tr>
<td>wbt_roberts_cross_filter</td>
<td>401</td>
</tr>
<tr>
<td>wbt_root_mean_square_error</td>
<td>402</td>
</tr>
<tr>
<td>wbtRotor</td>
<td>402</td>
</tr>
<tr>
<td>wbt_round</td>
<td>403</td>
</tr>
</tbody>
</table>
R topics documented:

wbt_ruggedness_index .................................................. 404
wbt_run_tool .......................................................... 405
wbt_scharr_filter ...................................................... 406
wbt_sediment_transport_index ................................. 407
wbt_select_tiles_by_polygon .................................... 408
wbt_set_nodata_value ............................................... 409
wbt_shadow_animation .............................................. 410
wbt_shadow_image .................................................. 411
wbt_shape_complexity_index ....................................... 412
wbt_shape_complexity_index_raster ............................... 413
wbt_shape_index ....................................................... 413
wbt_shreve_stream_magnitude ...................................... 414
wbt_sigmoidal_contrast_stretch ....................................... 415
wbt_sin ................................................................. 416
wbt_single_part_to_multi_part ................................... 417
wbt_sinh ............................................................... 418
wbt_sink ................................................................. 418
wbt_slope ............................................................... 419
wbt_slope_vs_aspect_plot ........................................... 420
wbt_slope_vs_elevation_plot ........................................ 421
wbt_smooth_vectors .................................................. 422
wbt_smooth_vegetation_residual ..................................... 423
wbt_snap_pour_points ................................................ 424
wbt_sobel_filter ....................................................... 425
wbt_spherical_std_dev_of_normals ................................. 426
wbt_split_colour_composite .......................................... 427
wbt_split_vector_lines ................................................ 428
wbt_split_with_lines ................................................ 429
wbt_square ............................................................... 430
wbt_square_root ....................................................... 430
wbt_standard_deviation_contrast_stretch ......................... 431
wbt_standard_deviation_filter ...................................... 432
wbt_standard_deviation_of_slope ................................... 433
wbt_stochastic_depression_analysis ................................. 434
wbt_strahler_order_basins ........................................... 435
wbt_strahler_stream_order ........................................... 436
wbt_stream_link_class ................................................ 437
wbt_stream_link_identifier .......................................... 438
wbt_stream_link_length ............................................. 439
wbt_stream_link_slope ............................................... 440
wbt_stream_power_index ........................................... 441
wbt_stream_slope_continuous ....................................... 442
wbt_subbasins .......................................................... 443
wbt_subtract ........................................................... 444
wbt_sum_overlay ........................................................ 445
wbt_surface_area_ratio ............................................... 445
wbt_svm_classification .............................................. 446
wbt_svm_regression .................................................. 447
wbt_symmetrical_difference ................................................. 449
wbt_tan ................................................................. 450
wbt_tangential_curvature ..................................................... 450
wbt_tanh ................................................................. 451
wbt_thicken_raster_line .................................................... 452
wbt_time_in_daylight ....................................................... 453
wbt_tin_gridding ........................................................... 454
wbt_toolbox ............................................................... 455
wbt_tool_help ............................................................. 456
wbt_tool_parameters ........................................................ 456
wbt_tophat_transform ....................................................... 457
wbt_topographic_position_animation .................................... 458
wbt_topological_stream_order ......................................... 459
wbt_total_curvature ....................................................... 460
wbt_total_filter ........................................................... 461
wbt_to_degrees ............................................................ 462
wbt_to_radians ............................................................. 462
wbt_trace_downslope_flowpaths ......................................... 463
wbt_trend_surface .......................................................... 464
wbt_trend_surface_vector_points ........................................ 465
wbt_tributary_identifier ................................................... 466
wbt_truncate ............................................................... 467
wbt_turning_bands_simulation ........................................... 468
wbt_two_sample_ks_test ................................................... 469
wbt_union ................................................................. 470
wbt_unnest_basins .......................................................... 471
wbt_unsharp_masking ....................................................... 472
wbt_unsphericity ............................................................. 473
wbt_update_nodata_cells .................................................. 474
wbt_upslope_depression_storage ......................................... 475
wbt_user_defined_weights_filter ....................................... 475
wbt_vector_hex_binning .................................................... 476
wbt_vector_lines_to_raster ............................................... 477
wbt_vector_points_to_raster ............................................. 478
wbt_vector_polygons_to_raster ......................................... 479
wbt_vector_stream_network_analysis .................................. 480
wbt_version ............................................................... 481
wbt_vertical_excess_curvature ........................................... 482
wbt_viewshed .............................................................. 483
wbt_view_code ............................................................ 484
wbt_visibility_index ....................................................... 484
wbt_voronoi_diagram ....................................................... 485
wbt_watershed ............................................................. 486
wbt_weighted_overlay ..................................................... 487
wbt_weighted_sum ........................................................ 488
wbt_wetness_index ........................................................ 489
wbt_wilcoxon_signed_rank_test ......................................... 490
wbt_write_function_memory_insertion ................................ 491
check_whitebox_binary

Description

Check for WhiteboxTools executable path

Usage

check_whitebox_binary(silent = TRUE)

Arguments

silent logical. Print help on installation/setting path. Default TRUE.

Value

logical if WhiteboxTools executable file exists.

See Also

wbt_exe_path()

sample_dem_data

Convenience method for path to sample DEM

Description

Get a file path to DEM.tif stored in extdata subfolder of whitebox package installation directory. If needed, download the TIFF file from GitHub.

Usage

sample_dem_data(
  destfile = file.path(system.file("extdata", package = "whitebox"), "DEM.tif"),
  ...
)
wbttoolparameters

Arguments

destfile
Path to target location of sample data. Will be downloaded if does not exist.
Defaults to file path of extdata subfolder of whitebox package installation directory.

... additional arguments to download.file()

Value
character.

Examples

if (check_whitebox_binary()) {
    wbt_slope(sample_dem_data(), output = "slope.tif")
}
unlink(c("slope.tif", 'settings.json'))

wbttoolparameters

WhiteBoxTools Tool Parameters

Description
This data set is a data.frame containing tools by name, their parameters, and associated metadata, as available in WhiteboxTools 1.5.0

Format
A data.frame with 1706 observations of 13 variables

- "function_name" - R function name
- "tool_name" - WhiteboxTools tool name
- "name" - parameter name
- "flags" - flags used to specify parameter on command line; comma separated
- "description" - parameter description
- "parameter_class" - parameter type
- "parameter_detail" - parameter details; character: data type followed by colon and more specifics, For OptionList possible values, comma-separated (if defined)
- "default_value" - parameter default value, if any
- "optional" - parameter "optional" flag; note that some combination of optional parameters may be required for certain conditions
- "label" - labels for selected subset of "flags" used as R function argument names for wbt_ functions
- "is_input" - logical. Classification of 'input' parameters
- "is_output" - logical. Classification of 'output' parameters
**Source**

WhiteboxTools 1.5.0

**See Also**

wbttools wbt_tool_parameters()

---

### wbttools

**WhiteboxTools Tool List**

**Description**

This data set is a data.frame containing tools by name and associated R function name, as available in WhiteboxTools 1.5.0

**Format**

A data.frame with 448 observations of 4 variables

- "tool_name" - WhiteboxTools tool name
- "toolbox_name" - WhiteboxTools toolbox name
- "description" - Brief description
- "function_name" - R function name

**Source**

WhiteboxTools 1.5.0

---

### wbt_absolute_value

**Absolute value**

**Description**

Calculates the absolute value of every cell in a raster.

**Usage**

```r
wbt_absolute_value(  
  input,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```
wbt_accumulation_curvature

Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

wbt_accumulation_curvature

*Accumulation curvature*

Description

This tool calculates accumulation curvature from an input DEM.

Usage

```r
wbt_accumulation_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **dem**: Name of the input raster DEM file.
- **output**: Name of the output raster image file.
- **log**: Display output values using a log-scale.
- **zfactor**: Z conversion factor.
- **wd**: Changes the working directory.
**wbt_adaptive_filter**

Sets verbose mode. If verbose mode is **FALSE**, tools will not print output messages.

Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

Return command that would be executed by `system()` rather than running tool.

Returns the tool text outputs.

---

**Description**

Performs an adaptive filter on an image.

**Usage**

```r
wbt_adaptive_filter(
  input,  # Input raster file.
  output,  # Output raster file.
  filterx = 11,  # Size of the filter kernel in the x-direction.
  filtery = 11,  # Size of the filter kernel in the y-direction.
  threshold = 2,  # Difference from mean threshold, in standard deviations.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is **FALSE**, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE  # Return command that would be executed by `system()` rather than running tool.
)
```

**Arguments**

- `input`  
  Input raster file.
- `output`  
  Output raster file.
- `filterx`  
  Size of the filter kernel in the x-direction.
- `filtery`  
  Size of the filter kernel in the y-direction.
- `threshold`  
  Difference from mean threshold, in standard deviations.
- `wd`  
  Changes the working directory.
- `verbose_mode`  
  Sets verbose mode. If verbose mode is **FALSE**, tools will not print output messages.
- `compress_rasters`  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`  
  Return command that would be executed by `system()` rather than running tool.
**Description**
Performs an addition operation on two rasters or a raster and a constant value.

**Usage**
```
wbt_add(
input1,
input2,
output,
wd = NULL,
verbose_mode = FALSE,
compress_rasters = FALSE,
command_only = FALSE
)
```

**Arguments**
- `input1`: Input raster file or constant value.
- `input2`: Input raster file or constant value.
- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**
Returns the tool text outputs.
### wbt_add_point_coordinates_to_table

*Add point coordinates to table*

**Description**

Modifies the attribute table of a point vector by adding fields containing each point’s X and Y coordinates.

**Usage**

```r
wbt_add_point_coordinates_to_table(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>input</code></td>
<td>Input vector Points file.</td>
</tr>
<tr>
<td><code>wd</code></td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td><code>verbose_mode</code></td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td><code>compress_rasters</code></td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td><code>command_only</code></td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.

### wbt_aggregate_raster

*Aggregate raster*

**Description**

Aggregates a raster to a lower resolution.
**Usage**

```r
wbt_aggregate_raster(
    input,  
    output,  
    agg_factor = 2,  
    type = "mean",  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE
)
```

**Arguments**

- `output`: Output raster file.
- `agg_factor`: Aggregation factor, in pixels.
- `type`: Statistic used to fill output pixels.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_and**

**And**

**Description**

Performs a logical AND operator on two Boolean raster images.

**Usage**

```r
wbt_and(
    input1,  
    input2,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,
```
wbt_anova

compress_rasters = FALSE,
command_only = FALSE
)

Arguments

input1 Input raster file.
input2 Input raster file.
output Output raster file.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Performs an analysis of variance (ANOVA) test on a raster dataset.

Usage

wbt_anova(
  input,
  features,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

- `features`: Feature definition (or class) raster.
- `output`: Output HTML file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

```r
wbt_arcosh
```

Description

Returns the inverse hyperbolic cosine (arcosh) of each values in a raster.

Usage

```r
wbt_arcosh(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.
Value

Returns the tool text outputs.

---

**Description**

Returns the inverse cosine (arccos) of each values in a raster.

**Usage**

```
wbt_arc_cos(
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE
)
```

**Arguments**

- **input**  
  Input raster file.

- **output**  
  Output raster file.

- **wd**  
  Changes the working directory.

- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

- **compress_rasters**  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

- **command_only**  
  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
Description

Returns the inverse sine (arcsin) of each values in a raster.

Usage

```r
wbt_arc_sin(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `input` Input raster file.
- `output` Output raster file.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Returns the inverse tangent (arctan) of each values in a raster.
Usage

wbt_arc_tan(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Returns the inverse hyperbolic sine (arsinh) of each values in a raster.
Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_artanh**

**Artanh**

Description

Returns the inverse hyperbolic tangent (arctanh) of each values in a raster.

Usage

```r
wbt_artanh(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.
Value

Returns the tool text outputs.

Description

Converts one or more ASCII files containing LiDAR points into LAS files.

Usage

```r
wbt_ascii_to_las(
  inputs,
  pattern,
  proj = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **inputs**: Input LiDAR ASCII files (.csv).
- **pattern**: Input field pattern.
- **proj**: Well-known-text string or EPSG code describing projection.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_aspect

Aspect

Description

Calculates an aspect raster from an input DEM.

Usage

wbt_aspect(
  dem, output, zfactor = NULL, wd = NULL, verbose_mode = FALSE,
  compress_rasters = FALSE, command_only = FALSE
)

Arguments

dem  Input raster DEM file.
output  Output raster file.
zfactor  Optional multiplier for when the vertical and horizontal units are not the same.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
**Description**

This tool assesses a route for slope, elevation, and visibility variation.

**Usage**

```r
wbt_assess_route(
  routes,
  dem,
  output,
  length = "",
  dist = 20,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **routes**: Name of the input routes vector file.
- **dem**: Name of the input DEM raster file.
- **output**: Name of the output lines shapefile.
- **length**: Maximum segment length (m).
- **dist**: Search distance, in grid cells, used in visibility analysis.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
Description

Returns the 2-argument inverse tangent (atan2).

Usage

```r
wbt_atan2(
  input_y,
  input_x,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input_y**: Input y raster file or constant value (rise).
- **input_x**: Input x raster file or constant value (run).
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_attribute_correlation

Attribute correlation

Description
Performs a correlation analysis on attribute fields from a vector database.

Usage
wbt_attribute_correlation(input, output = NULL, wd = NULL, verbose_mode = FALSE, compress_rasters = FALSE, command_only = FALSE)

Arguments
- **input**: Input vector file.
- **output**: Output HTML file (default name will be based on input file if unspecified).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

wbt_attribute_correlation_neighbourhood_analysis

Attribute correlation neighbourhood analysis

Description
Performs a correlation on two input vector attributes within a neighbourhood search windows.
Usage

wbt_attribute_correlation_neighbourhood_analysis(
    input,
    field1,
    field2,
    radius = NULL,
    min_points = NULL,
    stat = "pearson",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)  

Arguments

input Input vector file.
field1 First input field name (dependent variable) in attribute table.
field2 Second input field name (independent variable) in attribute table.
radius Search Radius (in map units).
min_points Minimum number of points.
stat Correlation type; one of 'pearson' (default) and 'spearman'.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_attribute_histogram

Attribute histogram

Description

Creates a histogram for the field values of a vector's attribute table.
Usage

```r
wbt_attribute_scattergram(
  input,
  fieldx,
  fieldy,
  output,
  trendline = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `fieldx`: Input field name in attribute table.
- `fieldy`: Output field name in attribute table.
- `output`: Output HTML file (default name will be based on input file if unspecified).
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Creating a scattergram for two field values of a vector’s attribute table.
Arguments

- **input**: Input raster file.
- **fieldx**: Input field name in attribute table for the x-axis.
- **fieldy**: Input field name in attribute table for the y-axis.
- **output**: Output HTML file (default name will be based on input file if unspecified).
- **trendline**: Draw the trendline.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Measures the average slope gradient from each grid cell to all upslope divide cells.

Usage

```r
wbt_average_flowpath_slope(
  dem,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
Arguments

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Calculates the circular variance of aspect at a scale for a DEM.

Usage

```r
wbt_average_normal_vector_angular_deviation(
  dem,
  output,
  filter = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **filter**: Size of the filter kernel.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.
**wbt_average_overlay**

**Value**

Returns the tool text outputs.

---

**Description**

Calculates the average for each grid cell from a group of raster images.

**Usage**

```r
wbt_average_overlay(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `inputs` Input raster files.
- `output` Output raster file.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_average_upslope_flowpath_length**

*Average upslope flowpath length*

**Description**

Measures the average length of all upslope flowpaths draining each grid cell.

**Usage**

```r
wbt_average_upslope_flowpath_length(dem, output, wd = NULL,
            verbose_mode = FALSE, compress_rasters = FALSE,
            command_only = FALSE)
```

**Arguments**

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_balance_contrast_enhancement**

*Balance contrast enhancement*

**Description**

Performs a balance contrast enhancement on a colour-composite image of multispectral data.
Usage

```r
wbt_balance_contrast_enhancement(
    input,
    output,
    band_mean = 100,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

Arguments

- **input**: Input colour composite image file.
- **output**: Output raster file.
- **band_mean**: Band mean value.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

### Basins

#### Description

Identifies drainage basins that drain to the DEM edge.

#### Usage

```r
wbt_basins(
    d8_pntr,
    output,
    esri_pntr = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```
**Arguments**

- `d8_pntr` Input raster D8 pointer file.
- `output` Output raster file.
- `esri_pntr` D8 pointer uses the ESRI style scheme.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**Description**

A bilateral filter is an edge-preserving smoothing filter introduced by Tomasi and Manduchi (1998).

**Usage**

```r
wbt_bilateral_filter(
  input,  # Input raster file.
  output,  # Output raster file.
  sigma_dist = 0.75,  # Standard deviation in distance in pixels.
  sigma_int = 1,  # Standard deviation in intensity in pixels.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE  # Return command that would be executed by system() rather than running tool.
)
```

**Arguments**

- `input` Input raster file.
- `output` Output raster file.
- `sigma_dist` Standard deviation in distance in pixels.
- `sigma_int` Standard deviation in intensity in pixels.
- `wd` Changes the working directory.
wbt_block_maximum_gridding

Block maximum gridding

Description

Creates a raster grid based on a set of vector points and assigns grid values using a block maximum scheme.

Usage

wbt_block_maximum_gridding(
  input,
  field,
  output,
  use_z = FALSE,
  cell_size = NULL,
  base = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input          Input vector Points file.
field          Input field name in attribute table.
output         Output raster file.
use_z          Use z-coordinate instead of field?.
cell_size      Optionally specified cell size of output raster. Not used when base raster is specified.
base           Optionally specified input base raster file. Not used when a cell size is specified.
wd              Changes the working directory.

verbose_mode   Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only   Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
wbt_block_minimum_gridding

verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

wbt_block_minimum_gridding

Block minimum gridding

Description
Creates a raster grid based on a set of vector points and assigns grid values using a block minimum scheme.

Usage

wbt_block_minimum_gridding(
  input,
  field,
  output,
  use_z = FALSE,
  cell_size = NULL,
  base = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input Input vector Points file.
field Input field name in attribute table.
output Output raster file.
use_z Use z-coordinate instead of field?.
cell_size Optionally specified cell size of output raster. Not used when base raster is specified.
base Optionally specified input base raster file. Not used when a cell size is specified.
wd Changes the working directory.
**wbt_boundary_shape_complexity**

| verbose_mode | Sets verbose mode. If verbose mode is FALSE, tools will not print output messages. |
| compress_rasters | Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters. |
| command_only | Return command that would be executed by `system()` rather than running tool. |

**Value**

Returns the tool text outputs.

---

**wbt_boundary_shape_complexity**

*Boundary shape complexity*

**Description**

Calculates the complexity of the boundaries of raster polygons.

**Usage**

```
wbt_boundary_shape_complexity(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
Description
Breaches all of the depressions in a DEM using Lindsay’s (2016) algorithm. This should be preferred over depression filling in most cases.

Usage

```r
wbt_breach_depressions(
  dem,
  output,
  max_depth = NULL,
  max_length = NULL,
  flat_increment = NULL,
  fill_pits = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

dem Input raster DEM file.
output Output raster file.
max_depth Optional maximum breach depth (default is Inf).
max_length Optional maximum breach channel length (in grid cells; default is Inf).
flat_increment Optional elevation increment applied to flat areas.
fill_pits Optional flag indicating whether to fill single-cell pits.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
wbt_breach_depressions_least_cost

Breach depressions least cost

Description

Breaches the depressions in a DEM using a least-cost pathway method.

Usage

wbt_breach_depressions_least_cost(
  dem,
  output,
  dist,
  max_cost = NULL,
  min_dist = TRUE,
  flat_increment = NULL,
  fill = TRUE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

dem                      Input raster DEM file.
output                   Output raster file.
dist                     Maximum search distance for breach paths in cells.
max_cost                 Optional maximum breach cost (default is Inf).
min_dist                 Optional flag indicating whether to minimize breach distances.
flat_increment           Optional elevation increment applied to flat areas.
fill                     Optional flag indicating whether to fill any remaining unbreached depressions.
wd                       Changes the working directory.
verbose_mode             Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters         Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only             Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
**wbt_breach_single_cell_pits**

*Breach single cell pits*

**Description**

Removes single-cell pits from an input DEM by breaching.

**Usage**

```r
wbt_breach_single_cell_pits(
  dem,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_buffer_raster**

*Buffer raster*

**Description**

Maps a distance-based buffer around each non-background (non-zero/non-nodata) grid cell in an input image.
Usage

```r
wbt_buffer_raster(
  input,
  output,
  size,
  gridcells = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `input` Input raster file.
- `output` Output raster file.
- `size` Buffer size.
- `gridcells` Optional flag to indicate that the ‘size’ threshold should be measured in grid cells instead of the default map units.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Burns-in streams at the sites of road embankments.
Usage

```r
wbt_burn_streams_at_roads(
  dem,
  streams,
  roads,
  output,
  width = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **dem**: Input raster digital elevation model (DEM) file.
- **streams**: Input vector streams file.
- **roads**: Input vector roads file.
- **output**: Output raster file.
- **width**: Maximum road embankment width, in map units.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

wbt_canny_edge_detection

*Canny edge detection*

Description

This tool performs a Canny edge-detection filter on an input image.
Usage

wbt_canny_edge_detection(
    input,
    output,
    sigma = 0.5,
    low = 0.05,
    high = 0.15,
    add_back = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input            Name of the input raster image file.
output           Name of the output raster image file.
sigma            Sigma value used in Gaussian filtering, default = 0.5.
low              Low threshold, default = 0.05.
high             High threshold, default = 0.15.
add_back         Add the edge cells back to the input image.
wd               Changes the working directory.
verbose_mode     Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Set the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only     Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt.ceil

Description

Returns the smallest (closest to negative infinity) value that is greater than or equal to the values in a raster.
Usage

wbt.ceil(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input       Input raster file.
output      Output raster file.
wd          Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output mes-
sages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression
for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Calculates the centroid, or average location, of raster polygon objects.

Usage

wbt_centroid(
  input,
  output,
  text_output = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input raster file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>text_output</td>
<td>Optional text output.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output</td>
</tr>
<tr>
<td></td>
<td>messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression</td>
</tr>
<tr>
<td></td>
<td>for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>

Value

Returns the tool text outputs.

---

wbt_centroid_vector  Centroid vector

Description

Identifies the centroid point of a vector polyline or polygon feature or a group of vector points.

Usage

wbt_centroid_vector(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input vector file.</td>
</tr>
<tr>
<td>output</td>
<td>Output vector file.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output</td>
</tr>
<tr>
<td></td>
<td>messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression</td>
</tr>
<tr>
<td></td>
<td>for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>
**wbt_change_vector_analysis**

**Value**

Returns the tool text outputs.

---

**wbt_change_vector_analysis**

*Change vector analysis*

**Description**

Performs a change vector analysis on a two-date multi-spectral dataset.

**Usage**

```r
wbt_change_vector_analysis(
  date1,
  date2,
  magnitude,
  direction,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **date1**: Input raster files for the earlier date.
- **date2**: Input raster files for the later date.
- **magnitude**: Output vector magnitude raster file.
- **direction**: Output vector Direction raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_circular_variance_of_aspect

Circular variance of aspect

Description

Calculates the circular variance of aspect at a scale for a DEM.

Usage

wbt_circular_variance_of_aspect(
  dem,
  output,
  filter = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

dem Input raster DEM file.
output Output raster file.
filter Size of the filter kernel.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**wbt_classify_buildings_in_lidar**

*Classify buildings in lidar*

**Description**

Reclassifies a LiDAR points that lie within vector building footprints.

**Usage**

```r
wbt_classify_buildings_in_lidar(
  input,
  buildings,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input`  
  Input LiDAR file.
- `buildings`  
  Input vector polygons file.
- `output`  
  Output LiDAR file.
- `wd`  
  Changes the working directory.
- `verbose_mode`  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`  
  Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.
wbt_classify_overlap_points

*Classify overlap points*

**Description**

Classifies or filters LAS points in regions of overlapping flight lines.

**Usage**

```r
wbt_classify_overlap_points(
  input,
  output,
  resolution = 2,
  filter = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input`: Input LiDAR file.
- `output`: Output LiDAR file.
- `resolution`: The size of the square area used to evaluate nearby points in the LiDAR data.
- `filter`: Filter out points from overlapping flightlines? If false, overlaps will simply be classified.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_clean_vector

**Clean vector**

Description

Removes null features and lines/polygons with fewer than the required number of vertices.

Usage

```r
wbt_clean_vector(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input vector file.
- **output**: Output vector file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

wbt_clip

**Clip**

Description

Extract all the features, or parts of features, that overlap with the features of the clip vector.
Usage

\begin{verbatim}
wbt_clip_lidar_to_polygon

Clip lidar to polygon

Description

Clips a LiDAR point cloud to a vector polygon or polygons.

Usage

\begin{verbatim}
wbt_clip_lidar_to_polygon(
    input,
    polygons,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
\end{verbatim}
\end{verbatim}

Arguments

- **input**: Input vector file.
- **clip**: Input clip polygon vector file.
- **output**: Output vector file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
Arguments

- input: Input LiDAR file.
- output: Output LiDAR file.
- wd: Changes the working directory.
- verbose_mode: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- compress_rasters: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- command_only: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Clips a raster to a vector polygon.

Usage

```r
describe_clipping(  
  input,  
  polygons,  
  output,  
  maintain_dimensions = FALSE,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

Arguments

- output: Output raster file.
- maintain_dimensions: Maintain input raster dimensions?
Description

A closing is a mathematical morphology operation involving an erosion (min filter) of a dilation (max filter) set.

Usage

```r
wbt_closing(
  input, output, filterx = 11, filtry = 11, wd = NULL, verbose_mode = FALSE, compress_rasters = FALSE, command_only = FALSE
)
```

Arguments

- `input` Input raster file.
- `output` Output raster file.
- `filterx` Size of the filter kernel in the x-direction.
- `filtry` Size of the filter kernel in the y-direction.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_clump

Value

Returns the tool text outputs.

wbt_clump  

Description

Groups cells that form discrete areas, assigning them unique identifiers.

Usage

wbt_clump(
  input,
  output,
  diag = TRUE,
  zero_back = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input  Input raster file.
output  Output raster file.
diag  Flag indicating whether diagonal connections should be considered.
zero_back  Flag indicating whether zero values should be treated as a background.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
wbt_compactness_ratio  Compactness ratio

Description

Calculates the compactness ratio (A/P), a measure of shape complexity, for vector polygons.

Usage

wbt_compactness_ratio(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input  Input vector polygon file.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbtConditionalEvaluation  Conditional evaluation

Description

This tool performs a conditional evaluation (if-then-else) operation on a raster.
Usage

```r
wbt_conditional_evaluation(
    input,
    output,
    statement = "",
    true = NULL,
    false = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

Arguments

- **input**: Name of the input raster file.
- **output**: Name of the output raster file.
- **statement**: Conditional statement e.g. value > 35.0. This statement must be a valid Rust statement.
- **true**: Value where condition evaluates TRUE (input raster or constant value).
- **false**: Value where condition evaluates FALSE (input raster or constant value).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Perform a conservative-smoothing filter on an image.
Usage

```
wbt_conservative_smoothing_filter(
    input,  
    output,  
    filterx = 3,  
    filtery = 3,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input raster file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>filterx</td>
<td>Size of the filter kernel in the x-direction.</td>
</tr>
<tr>
<td>filtery</td>
<td>Size of the filter kernel in the y-direction.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>

Value

Returns the tool text outputs.

---

**wbt_construct_vector_tin**

*Construct vector tin*

**Description**

Creates a vector triangular irregular network (TIN) for a set of vector points.

**Usage**

```
wbt_construct_vector_tin(
    input,  
    output,  
    field = NULL,  
    use_z = FALSE,
```
wbt_contours_from_points

Arguments

- **input**: Input vector points file.
- **output**: Output vector polygon file.
- **field**: Input field name in attribute table.
- **use_z**: Use the 'z' dimension of the Shapefile's geometry instead of an attribute field?
- **max_triangle_edge_length**: Optional maximum triangle edge length; triangles larger than this size will not be gridded.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Creates a contour coverage from a set of input points.

Usage

```r
wbt_contours_from_points(
  input, output,
  field = NULL, use_z = FALSE,
  max_triangle_edge_length = NULL, interval = 10,
)```

Contours from points

```r
wbt_contours_from_points
```
Arguments

input  Input vector points file.
output Output vector lines file.
field  Input field name in attribute table.
use_z Use the 'z' dimension of the Shapefile's geometry instead of an attribute field?.
max_triangle_edge_length  Optional maximum triangle edge length; triangles larger than this size will not be gridded.
interval Contour interval.
base Base contour height.
smooth Smoothing filter size (in num. points), e.g. 3, 5, 7, 9, 11.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Derives a vector contour coverage from a raster surface.
Usage

wbt_contours_from_raster(
    input,
    output,
    interval = 10,
    base = 0,
    smooth = 9,
    tolerance = 10,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input Input surface raster file.
output Output vector contour file.
interval Contour interval.
base Base contour height.
smooth Smoothing filter size (in num. points), e.g. 3, 5, 7, 9, 11.
tolerance Tolerance factor, in degrees (0-45); determines generalization level.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_convert_nodata_to_zero

Convert nodata to zero

Description

Converts nodata values in a raster to zero.
Usage

wbt_convert_nodata_to_zero(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input  Input raster file.
output  Output raster file.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

wbt_convert_raster_format

Convert raster format

Description

Converts raster data from one format to another.

Usage

wbt_convert_raster_format(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
wbt_corner_detection

Arguments

input       Input raster file.
output      Output raster file.
wd          Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_corner_detection  Corner detection

Description

Identifies corner patterns in boolean images using hit-and-miss pattern matching.

Usage

wbt_corner_detection(
  input,           
  output,          
  wd = NULL,       
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)

Arguments

input       Input boolean image.
output      Output raster file.
wd          Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.
**Description**

Corrects the darkening of images towards corners.

**Usage**

```r
wbt_correct_vignetting(
  input,
  pp,
  output,
  focal_length = 304.8,
  image_width = 228.6,
  n = 4,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **pp**: Input principal point file.
- **output**: Output raster file.
- **focal_length**: Camera focal length, in millimeters.
- **image_width**: Distance between photograph edges, in millimeters.
- **n**: The 'n' parameter.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_cos**

**Cos**

**Description**

Returns the cosine (cos) of each values in a raster.

**Usage**

```r
wbt_cos(
    input,  # Input raster file.
    output, # Output raster file.
    wd = NULL,   # Changes the working directory.
    verbose_mode = FALSE,   # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
    compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
    command_only = FALSE  # Return command that would be executed by system() rather than running tool.
)
```

**Arguments**

- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_cosh**

**Cosh**

**Description**

Returns the hyperbolic cosine (cosh) of each values in a raster.
Usage

wbt_cosh(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input  Input raster file.
output  Output raster file.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_cost_allocation  Cost allocation

Description

Identifies the source cell to which each grid cell is connected by a least-cost pathway in a cost-distance analysis.

Usage

wbt_cost_allocation(
    source,
    backlink,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
**Arguments**

- **source**: Input source raster file.
- **backlink**: Input backlink raster file generated by the cost-distance tool.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_cost_distance**  
*Cost distance*

**Description**

Performs cost-distance accumulation on a cost surface and a group of source cells.

**Usage**

```r
wbt_cost_distance(
  source,
  cost,
  out_accum,
  out_backlink,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **source**: Input source raster file.
- **cost**: Input cost (friction) raster file.
- **out_accum**: Output cost accumulation raster file.
- **out_backlink**: Output backlink raster file.
- **wd**: Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only  Return command that would be executed by system() rather than running tool.

Value  Returns the tool text outputs.

---

**Description**

Performs cost-distance pathway analysis using a series of destination grid cells.

**Usage**

```r
wbt_cost_pathway(
  destination,
  backlink,
  output,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **destination**  Input destination raster file.
- **backlink**  Input backlink raster file generated by the cost-distance tool.
- **output**  Output cost pathway raster file.
- **zero_background**  Flag indicating whether zero values should be treated as a background.
- **wd**  Changes the working directory.
- **verbose_mode**  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**  Return command that would be executed by system() rather than running tool.
**wbt_count_if**

**Value**

Returns the tool text outputs.

**Description**

Counts the number of occurrences of a specified value in a cell-stack of rasters.

**Usage**

```r
wbt_count_if(
  inputs,
  output,
  value,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **inputs**: Input raster files.
- **output**: Output raster file.
- **value**: Search value (e.g. `countif value = 5.0`).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_create_colour_composite

Create colour composite

Description

Creates a colour-composite image from three bands of multispectral imagery.

Usage

```r
wbt_create_colour_composite(
  red,
  green,
  blue,
  output,
  opacity = NULL,
  enhance = TRUE,
  zeros = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `red`: Input red band image file.
- `green`: Input green band image file.
- `blue`: Input blue band image file.
- `output`: Output colour composite file.
- `opacity`: Input opacity band image file (optional).
- `enhance`: Optional flag indicating whether a balance contrast enhancement is performed.
- `zeros`: Optional flag to indicate if zeros are nodata values.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
Create hexagonal vector grid

Description

Creates a hexagonal vector grid.

Usage

wbt_create_hexagonal_vector_grid(
  input,
  output,
  width,
  orientation = "horizontal",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input          Input base file.
output         Output vector polygon file.
width          The grid cell width.
orientation    Grid Orientation, ‘horizontal’ or ‘vertical’.
wd             Changes the working directory.
verbose_mode   Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only   Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
wbt_create_plane  

Create plane

Description

Creates a raster image based on the equation for a simple plane.

Usage

wbt_create_plane(
    base,  
    output,  
    gradient = 15,  
    aspect = 90,  
    constant = 0,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE
)

Arguments

base                      Input base raster file.
output                    Output raster file.
gradient                  Slope gradient in degrees (-85.0 to 85.0).
aspect                    Aspect (direction) in degrees clockwise from north (0.0-360.0).
constant                  Constant value.
wd                        Changes the working directory.
verbose_mode              Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters          Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only              Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
Description

Creates a rectangular vector grid.

Usage

```r
wbt_create_rectangular_vector_grid(
  input, output, width, height, xorig = 0, yorig = 0, wd = NULL,
  verbose_mode = FALSE, compress_rasters = FALSE, command_only = FALSE
)
```

Arguments

- **input**: Input base file.
- **output**: Output vector polygon file.
- **width**: The grid cell width.
- **height**: The grid cell height.
- **xorig**: The grid origin x-coordinate.
- **yorig**: The grid origin y-coordinate.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
Description

Calculates the Crispness Index, which is used to quantify how crisp (or conversely how fuzzy) a probability image is.

Usage

```r
wbt_crispness_index(
  input,
  output = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input raster file.
- **output**: Optional output html file (default name will be based on input file if unspecified).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Performs a cross-tabulation on two categorical images.
**wbt_csv_points_to_vector**

*Csv points to vector*

**Usage**

```r
wbt_csv_points_to_vector(
  input,  
  output,  
  xfield = 0,
  yfield = 1,  
  epsg = NULL,  
  wd = NULL,
)```

**Arguments**

- **input**: Input raster file 1.
- **input2**: Input raster file 1.
- **output**: Output HTML file (default name will be based on input file if unspecified).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**Description**

Converts a CSV text file to vector points.

**Usage**

```r
wbt_csv_points_to_vector(
  input,  
  output,  
  xfield = 0,
  yfield = 1,  
  epsg = NULL,  
  wd = NULL,
)```
Arguments

- **input**: Input CSV file (i.e. source of data to be imported).
- **output**: Output vector file.
- **xfield**: X field number (e.g. 0 for first field).
- **yfield**: Y field number (e.g. 1 for second field).
- **epsg**: EPSG projection (e.g. 2958).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_cumulative_distribution**

*Cumulative distribution*

**Description**

Converts a raster image to its cumulative distribution function.

**Usage**

```r
wbt_cumulative_distribution(
  input, output, wd = NULL,
  verbose_mode = FALSE, compress_rasters = FALSE, command_only = FALSE
)
```
**wbt_curvedness**  

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_curvedness**  

**Curvedness**

---

**Description**

This tool calculates curvedness from an input DEM.

**Usage**

```r
curvature <- function(dem, output, log = FALSE, zfactor = 1, wd = NULL, verbose_mode = FALSE, compress_rasters = FALSE, command_only = FALSE)
```

**Arguments**

- **dem**: Name of the input raster DEM file.
- **output**: Name of the output raster image file.
- **log**: Display output values using a log-scale.
- **zfactor**: Z conversion factor.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
compress_rasters

Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only

Return command that would be executed by `system()` rather than running tool.

### Value

Returns the tool text outputs.

---

**wbt_d8_flow_accumulation**

*D8 flow accumulation*

---

**Description**

Calculates a D8 flow accumulation raster from an input DEM or flow pointer.

**Usage**

```r
wbt_d8_flow_accumulation(
  input,
  output,
  out_type = "cells",
  log = FALSE,
  clip = FALSE,
  pntr = FALSE,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster DEM or D8 pointer file.
- **output**: Output raster file.
- **out_type**: Output type; one of 'cells' (default), 'catchment area', and 'specific contributing area'.
- **log**: Optional flag to request the output be log-transformed.
- **clip**: Optional flag to request clipping the display max by 1 percent.
- **pntr**: Is the input raster a D8 flow pointer rather than a DEM?
- **esri_pntr**: Input D8 pointer uses the ESRI style scheme.
- **wd**: Changes the working directory.
wbt_d8_mass_flux

verbose_mode
Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

---

wbt_d8_mass_flux  
*D8 mass flux*

**Description**
Performs a D8 mass flux calculation.

**Usage**

```r
wbt_d8_mass_flux(
  dem,          
  loading,      
  efficiency,   
  absorption,   
  output,       
  wd = NULL,    
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE 
)
```

**Arguments**

dem  
Input raster DEM file.

loading  
Input loading raster file.

efficiency  
Input efficiency raster file.

absorption  
Input absorption raster file.

output  
Output raster file.

wd  
Changes the working directory.

verbose_mode  
Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters  
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only  
Return command that would be executed by system() rather than running tool.
Returns the tool text outputs.
**Description**

Performs a DBSCAN-based unsupervised clustering operation.

**Usage**

```r
wbt_dbscan(
  inputs,
  output,
  scaling = "Normalize",
  search_dist = 0.01,
  min_points = 5,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `inputs`  
  Names of the input rasters.
- `output`  
  Name of the output raster file.
- `scaling`  
  Scaling method for predictors. Options include 'None', 'Normalize', and 'Standardize'.
- `search_dist`  
  Search-distance parameter.
- `min_points`  
  Minimum point density needed to define 'core' point in cluster.
- `wd`  
  Changes the working directory.
- `verbose_mode`  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`  
  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
### wbt_decrement

**Decrement**

Decreases the values of each grid cell in an input raster by 1.0 (see also InPlaceSubtract).

**Usage**

```r
wbt_decrement(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

### wbt_depth_in_sink

**Depth in sink**

Measures the depth of sinks (depressions) in a DEM.
**Usage**

```r
wbt_depth_in_sink(
  dem,
  output,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `output`: Output raster file.
- `zero_background`: Flag indicating whether the background value of zero should be used.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_dev_from_mean_elev**

*Dev from mean elev*

**Description**

Calculates deviation from mean elevation.

**Usage**

```r
wbt_dev_from_mean_elev(
  dem,
  output,
  filterx = 11,
  filtery = 11,
  wd = NULL,
)```
Verbose mode = FALSE, compress_rasters = FALSE, command_only = FALSE

Arguments

- output: Output raster file.
- filterx: Size of the filter kernel in the x-direction.
- filtery: Size of the filter kernel in the y-direction.
- wd: Changes the working directory.
- verbose_mode: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- compress_rasters: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- command_only: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_difference**

**Difference**

**Description**

Outputs the features that occur in one of the two vector inputs but not both, i.e. no overlapping features.

**Usage**

```r
wbt_difference(
  input,
  overlay,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
**Arguments**

- **input**: Input vector file.
- **overlay**: Input overlay vector file.
- **output**: Output vector file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_difference_curvature**

_Difference curvature_

---

**Description**

This tool calculates difference curvature from an input DEM.

**Usage**

```r
wbt_difference_curvature(
  dem, 
  output, 
  log = FALSE, 
  zfactor = 1, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE 
)
```

**Arguments**

- **dem**: Name of the input raster DEM file.
- **output**: Name of the output raster image file.
- **log**: Display output values using a log-scale.
- **zfactor**: Z conversion factor.
- **wd**: Changes the working directory.
Verbose mode
Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

Compress rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

Command only
Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

---

**wbt_diff_from_mean_elev**

*Diff from mean elev*

**Description**
Calculates difference from mean elevation (equivalent to a high-pass filter).

**Usage**

```r
wbt_diff_from_mean_elev(
  dem,
  output,
  filterx = 11,
  filtery = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem Input raster DEM file.
output Output raster file.
filterx Size of the filter kernel in the x-direction.
filtery Size of the filter kernel in the y-direction.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.
**wbt_diff_of_gaussian_filter**

**Value**

Returns the tool text outputs.

---

**Description**

Performs a Difference of Gaussian (DoG) filter on an image.

**Usage**

```r
wbt_diff_of_gaussian_filter(
  input,
  output,
  sigma1 = 2,
  sigma2 = 4,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `output`: Output raster file.
- `sigma1`: Standard deviation distance in pixels.
- `sigma2`: Standard deviation distance in pixels.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
Directional relief

Description

Calculates relief for cells in an input DEM for a specified direction.

Usage

```r
wbt_directional_relief(
  dem,
  output,
  azimuth = 0,
  max_dist = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

dem          Input raster DEM file.
output       Output raster file.
azimuth      Wind azimuth in degrees.
max_dist     Optional maximum search distance (unspecified if none; in xy units).
wd           Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
**Description**

Performs a direct decorrelation stretch enhancement on a colour-composite image of multispectral data.

**Usage**

```r
wbt_direct_decorrelation_stretch(
  input,
  output,
  k = 0.5,
  clip = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input colour composite image file.
- **output**: Output raster file.
- **k**: Achromatic factor (k) ranges between 0 (no effect) and 1 (full saturation stretch), although typical values range from 0.3 to 0.7.
- **clip**: Optional percent to clip the upper tail by during the stretch.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_dissolve**  

**Dissolve**

**Description**

Removes the interior, or shared, boundaries within a vector polygon coverage.

**Usage**

```r
wbt_dissolve(
  input,
  output,
  field = NULL,
  snap = 0,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input vector file.
- **output**: Output vector file.
- **field**: Dissolve field attribute (optional).
- **snap**: Snap tolerance.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_distance_to_outlet**

*Distance to outlet*

**Description**

Calculates the distance of stream grid cells to the channel network outlet cell.

**Usage**

```r
wbt_distance_to_outlet(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `d8_pntr` Input raster D8 pointer file.
- `streams` Input raster streams file.
- `output` Output raster file.
- `esri_pntr` D8 pointer uses the ESRI style scheme.
- `zero_background` Flag indicating whether a background value of zero should be used.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_diversity_filter**  
*Diversity filter*

**Description**
Assigns each cell in the output grid the number of different values in a moving window centred on each grid cell in the input raster.

**Usage**
```
wbt_diversity_filter(
  input,
  output,
  filterx = 11,
  filtery = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**
- **input**: Input raster file.
- **output**: Output raster file.
- **filterx**: Size of the filter kernel in the x-direction.
- **filtery**: Size of the filter kernel in the y-direction.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**
Returns the tool text outputs.
Description

Performs a division operation on two rasters or a raster and a constant value.

Usage

```r
wbt_divide(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input1**: Input raster file or constant value.
- **input2**: Input raster file or constant value.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_downslope_distance_to_stream

*Downslope distance to stream*

**Description**

Measures distance to the nearest downslope stream cell.

**Usage**

```r
wbt_downslope_distance_to_stream(
  dem,
  streams,
  output,
  dinf = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dem</td>
<td>Input raster DEM file.</td>
</tr>
<tr>
<td>streams</td>
<td>Input raster streams file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>dinf</td>
<td>Use the D-infinity flow algorithm instead of D8?.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.
Description

Calculates the downslope flowpath length from each cell to basin outlet.

Usage

```r
wbt_downslope_flowpath_length(
  d8_pntr,
  output,
  watersheds = NULL,
  weights = NULL,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `output`: Output raster file.
- `watersheds`: Optional input watershed raster file.
- `weights`: Optional input weights raster file.
- `esri_pntr`: D8 pointer uses the ESRI style scheme.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_downslope_index  
*Downslope index*

**Description**

Calculates the Hjerdt et al. (2004) downslope index.

**Usage**

```r
wbt_downslope_index(
  dem,
  output,
  drop = 2,
  out_type = "tangent",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **drop**: Vertical drop value (default is 2.0).
- **out_type**: Output type, options include 'tangent', 'degrees', 'radians', 'distance' (default is 'tangent').
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**Description**

Calculates a D-infinity flow accumulation raster from an input DEM.

**Usage**

```r
wbt_d_inf_flow_accumulation(
  input,  # Input raster DEM or D-infinity pointer file.
  output, # Output raster file.
  out_type = "Specific Contributing Area", # Output type; one of 'cells', 'sca' (default), and 'ca'.
  threshold = NULL,  # Optional convergence threshold parameter, in grid cells; default is infinity.
  log = FALSE,  # Optional flag to request the output be log-transformed.
  clip = FALSE,  # Optional flag to request clipping the display max by 1 percent.
  pntr = FALSE,  # Is the input raster a D-infinity flow pointer rather than a DEM?.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE  # Return command that would be executed by system() rather than running tool.
)
```

**Arguments**

- `input`: Input raster DEM or D-infinity pointer file.
- `output`: Output raster file.
- `out_type`: Output type; one of ‘cells’, ‘sca’ (default), and ‘ca’.
- `threshold`: Optional convergence threshold parameter, in grid cells; default is infinity.
- `log`: Optional flag to request the output be log-transformed.
- `clip`: Optional flag to request clipping the display max by 1 percent.
- `pntr`: Is the input raster a D-infinity flow pointer rather than a DEM?.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.
wbt_d_inf_mass_flux  

\textit{D inf mass flux}

**Description**

Performs a D-infinity mass flux calculation.

**Usage**

```
wbt_d_inf_mass_flux(
    dem,
    loading,
    efficiency,
    absorption,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

- **dem**: Input raster DEM file.
- **loading**: Input loading raster file.
- **efficiency**: Input efficiency raster file.
- **absorption**: Input absorption raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by \texttt{system()} rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_d_inf_pointer**  

**D inf pointer**

**Description**

Calculates a D-infinity flow pointer (flow direction) raster from an input DEM.

**Usage**

```r
wbt_d_inf_pointer(
  dem,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

**wbt_edge_contamination**

**Edge contamination**

**Description**

This tool identifies grid cells within an input DEM that may be impacted by edge contamination for hydrological applications.
**Usage**

```r
wbt_edge_contamination(
  dem,
  output,
  flow_type = "mfd",
  zfactor = "",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `dem` Name of the input DEM raster file; must be depressionless.
- `output` Name of the output raster file.
- `flow_type` Flow algorithm type, one of 'd8', 'mfd', or 'dinf'.
- `zfactor` Optional multiplier for when the vertical and horizontal units are not the same.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If `verbose_mode` is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_edge_density**

**Edge density**

**Description**

Calculates the density of edges, or breaks-in-slope within DEMs.

**Usage**

```r
wbt_edge_density(
  dem,
  output,
  filter = 11,
  norm_diff = 5,
  zfactor = NULL,
)```
wbt_edge_preserving_mean_filter

Edge preserving mean filter

Description

Performs a simple edge-preserving mean filter on an input image.

Usage

wbt_edge_preserving_mean_filter(
    input,
    output,
    threshold,
    filter = 11,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
Arguments

- **input**  
  Input raster file.
- **output**  
  Output raster file.
- **threshold**  
  Maximum difference in values.
- **filter**  
  Size of the filter kernel.
- **wd**  
  Changes the working directory.
- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**  
  Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

```r
wbt_edge_proportion  Edge proportion
```

Description

Calculate the proportion of cells in a raster polygon that are edge cells.

Usage

```r
wbt_edge_proportion(
  input, output,
  output_text = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**  
  Input raster file.
- **output**  
  Output raster file.
- **output_text**  
  Flag indicating whether a text report should also be output.
- **wd**  
  Changes the working directory.
- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.


**wbt_elevation_above_stream**

**Description**

Calculates the elevation of cells above the nearest downslope stream cell.

**Usage**

```r
wbt_elevation_above_stream(
  dem,
  streams,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `streams`: Input raster streams file.
- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_elevation_above_stream_euclidean**

*Elevation above stream euclidean*

**Description**

Calculates the elevation of cells above the nearest (Euclidean distance) stream cell.

**Usage**

```r
wbt_elevation_above_stream_euclidean(
  dem,
  streams,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **dem**: Input raster DEM file.
- **streams**: Input raster streams file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_elev_above_pit**  
*Elev above pit*

**Description**

Calculate the elevation of each grid cell above the nearest downstream pit cell or grid edge cell.

**Usage**

```r
wbt_elev_above_pit(
  dem,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_elev_percentile**  
*Elev percentile*

**Description**

Calculates the elevation percentile raster from a DEM.
wbt_elev_relative_to_min_max

Usage

```r
wbt_elev_percentile(
  dem,
  output,
  filterx = 11,
  filtery = 11,
  sig_digits = 2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

dem Input raster DEM file.
output Output raster file.
filterx Size of the filter kernel in the x-direction.
filtery Size of the filter kernel in the y-direction.
sig_digits Number of significant digits.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Calculates the elevation of a location relative to the minimum and maximum elevations in a DEM.
Usage

wbt_elev_relative_to_watershed_min_max(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

dem Input raster DEM file.
output Output raster file.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Calculates the elevation of a location relative to the minimum and maximum elevations in a watershed.

Usage

wbt_elev_relative_to_watershed_min_max(
    dem,
    watersheds,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
wbt_eliminate_coincident_points

Arguments

watersheds: Input raster watersheds file.
output: Output raster file.
wd: Changes the working directory.
verbose_mode: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_eliminate_coincident_points

Eliminate coincident points

Description

Removes any coincident, or nearly coincident, points from a vector points file.

Usage

wbt_eliminate_coincident_points(
  input,
  output,
  tolerance,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input: Input vector file.
output: Output vector polygon file.
tolerance: The distance tolerance for points.
wd: Changes the working directory.
verbose_mode: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
`wbt_elongation_ratio`  

**compress_rasters**  
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

**command_only**  
Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**`wbt_elongation_ratio`  
*Elongation ratio***

**Description**

Calculates the elongation ratio for vector polygons.

**Usage**

```r
wbt_elongation_ratio(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**  
  Input vector polygon file.

- **wd**  
  Changes the working directory.

- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

- **compress_rasters**  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

- **command_only**  
  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_embankment_mapping

Embankment mapping

Description
Maps and/or removes road embankments from an input fine-resolution DEM.

Usage

```r
wbt_embankment_mapping(
  dem,
  road_vec,
  output,
  search_dist = 2.5,
  min_road_width = 6,
  typical_width = 30,
  max_height = 2,
  max_width = 60,
  max_increment = 0.05,
  spillout_slope = 4,
  remove_embankments = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

dem
Input raster DEM file.

road_vec
Input vector polygons file.

output
Output raster file.

search_dist
Search distance used to reposition transportation vectors onto road embankments (in map units).

min_road_width
Minimum road width; this is the width of the paved road surface (in map units).

typical_width
Typical embankment width; this is the maximum width of an embankment with roadside ditches (in map units).

max_height
Typical embankment maximum height; this is the height a typical embankment with roadside ditches (in map units).

max_width
Maximum embankment width, typically where embankments traverse steep-sided valleys (in map units).

max_increment
Maximum upwards increment between neighbouring cells on an embankment (in elevation units).
**wbt_emboss_filter**

**spillout_slope**  Spillout slope (in degrees).

**remove_embankments**  Optional flag indicating whether to output a DEM with embankments removed (true) or an embankment raster map (false).

**wd**  Changes the working directory.

**verbose_mode**  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

**compress_rasters**  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

**command_only**  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_emboss_filter**  Emboss filter

---

**Description**

Performs an emboss filter on an image, similar to a hillshade operation.

**Usage**

```r
wbt_emboss_filter(
  input,
  output,
  direction = "n",
  clip = 0,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**  Input raster file.
- **output**  Output raster file.
- **direction**  Direction of reflection; options include 'n', 's', 'e', 'w', 'ne', 'se', 'nw', 'sw'.
- **clip**  Optional amount to clip the distribution tails by, in percent.
- **wd**  Changes the working directory.
- **verbose_mode**  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
`wbt_equal_to`  

Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

`command_only`  

Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**Description**

Performs a equal-to comparison operation on two rasters or a raster and a constant value.

**Usage**

```r
wbt_equal_to(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input1`  
  Input raster file or constant value.
- `input2`  
  Input raster file or constant value.
- `output`  
  Output raster file.
- `wd`  
  Changes the working directory.
- `verbose_mode`  
  Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`  
  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**Description**

Removes all the features, or parts of features, that overlap with the features of the erase vector polygon.

**Usage**

```r
wbt_erase(
  input,
  erase,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `output` : Output vector file.
- `wd` : Changes the working directory.
- `verbose_mode` : Sets verbose mode. If `verbose_mode` is `FALSE`, tools will not print output messages.
- `compress_rasters` : Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` : Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_erase_polygon_from_lidar

Erase polygon from lidar

Description

Erases (cuts out) a vector polygon or polygons from a LiDAR point cloud.

Usage

```r
wbt_erase_polygon_from_lidar(
  input,
  polygons,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input LiDAR file.
- **polygons**: Input vector polygons file.
- **output**: Output LiDAR file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**wbt_erase_polygon_from_raster**

*Erase polygon from raster*

**Description**

Erases (cuts out) a vector polygon from a raster.

**Usage**

```r
wbt_erase_polygon_from_raster(
  input,  
  polygons,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **polygons**: Input vector polygons file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_euclidean_allocation

Euclidean allocation

Description

Assigns grid cells in the output raster the value of the nearest target cell in the input image, measured by the Shih and Wu (2004) Euclidean distance transform.

Usage

wbt_euclidean_allocation(
  input,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)

Arguments

input  
Input raster file.

output  
Output raster file.

wd  
Changes the working directory.

verbose_mode  
Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters  
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only  
Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_euclidean_distance

Euclidean distance

Description

Usage

wbt_evaluate_training_sites(
    inputs,  # Input raster file.
    polys,  # Output raster file.
    field,  # Changes the working directory.
    output,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
    wd = NULL,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
    verbose_mode = FALSE,  # Return command that would be executed by system() rather than running tool.
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input Input raster file.
output Output raster file.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

This tool can be used to inspect the overlap in spectral signatures of training sites for various classes.

Usage

wbt_evaluate_training_sites(
    inputs,  # Input raster file.
    polys,  # Output raster file.
    field,  # Changes the working directory.
    output,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
    wd = NULL,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
    verbose_mode = FALSE,  # Return command that would be executed by system() rather than running tool.
    compress_rasters = FALSE,
    command_only = FALSE
)
Arguments

- **inputs**: Name of the input band images.
- **polys**: Name of the input training site polygons shapefile.
- **field**: Name of the attribute containing class name data.
- **output**: Name of the output report file (*.html).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

```r
wbt_exp
```

Description

Returns the exponential (base e) of values in a raster.

Usage

```r
wbt_exp(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.
Value

Returns the tool text outputs.

---

**wbt_exp2**

**Description**

Returns the exponential (base 2) of values in a raster.

**Usage**

```r
wbt_exp2(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_export_table_to_csv

*Export table to csv*

**Description**
Exports an attribute table to a CSV text file.

**Usage**

```r
wbt_export_table_to_csv(
    input,
    output,
    headers = TRUE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

- `input`  
  Input vector file.

- `output`  
  Output csv file.

- `headers`  
  Export field names as file header?.

- `wd`  
  Changes the working directory.

- `verbose_mode`  
  Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.

- `compress_rasters`  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

- `command_only`  
  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_exposure_towards_wind_flux

Exposure towards wind flux

Description

This tool evaluates hydrologic connectivity within a DEM.

Usage

```r
wbt_exposure_towards_wind_flux(
  dem,
  output,
  azimuth = "",
  max_dist = "",
  zfactor = "",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **dem**: Name of the input DEM raster file.
- **output**: Name of the output raster file.
- **azimuth**: Wind azimuth, in degrees.
- **max_dist**: Optional maximum search distance. Minimum value is 5 x cell size.
- **zfactor**: Optional multiplier for when the vertical and horizontal units are not the same.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_extend_vector_lines

*Extend vector lines*

**Description**

Extends vector lines by a specified distance.

**Usage**

```r
wbt_extend_vector_lines(
  input,
  output,
  dist,
  extend = "both ends",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input vector polyline file.</td>
</tr>
<tr>
<td>output</td>
<td>Output vector polyline file.</td>
</tr>
<tr>
<td>dist</td>
<td>The distance to extend.</td>
</tr>
<tr>
<td>extend</td>
<td>Extend direction, 'both ends' (default), 'line start', 'line end'.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.
**Description**

Converts vector lines or polygons into vertex points.

**Usage**

```r
wbt_extract_nodes(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input`: Input vector lines or polygon file.
- `output`: Output vector points file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

**Description**

Extracts the values of raster(s) at vector point locations.
Usage

wbt_extract_raster_values_at_points(
    inputs,
    points,
    out_text = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

inputs          Input raster files.
points          Input vector points file.
out_text        Output point values as text? Otherwise, the only output is to the points file’s attribute table.
wd              Changes the working directory.
verbose_mode    Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only    Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Extracts stream grid cells from a flow accumulation raster.

Usage

wbt_extract_streams(
    flow_accum,
    output,
    threshold,
    zero_background = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
Arguments

- **flow_accum**: Input raster D8 flow accumulation file.
- **output**: Output raster file.
- **threshold**: Threshold in flow accumulation values for channelization.
- **zero_background**: Flag indicating whether a background value of zero should be used.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_extract_valleys**  
*Extract valleys*

Description

Identifies potential valley bottom grid cells based on local topography alone.

Usage

```r
wbt_extract_valleys(
  dem,
  output,
  variant = "LQ",
  line_thin = TRUE,
  filter = 5,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
Arguments

dem Input raster DEM file.
output Output raster file.
variant Options include 'LQ' (lower quartile), 'JandR' (Johnston and Rosenfeld), and 'PandD' (Peucker and Douglas); default is 'LQ'.
line_thin Optional flag indicating whether post-processing line-thinning should be performed.
filter Optional argument (only used when variant='lq') providing the filter size, in grid cells, used for lq-filtering (default is 5).
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

wbt_farthest_channel_head

Farthest channel head

Description

Calculates the distance to the furthest upstream channel head for each stream cell.

Usage

```r
wbt_farthest_channel_head(
d8_pntr,
streams,
output,
esri_pntr = FALSE,
zero_background = FALSE,
wd = NULL,
verbose_mode = FALSE,
compress_rasters = FALSE,
command_only = FALSE
)
```
**Arguments**

- **d8_pntr**: Input raster D8 pointer file.
- **streams**: Input raster streams file.
- **output**: Output raster file.
- **esri_pntr**: D8 pointer uses the ESRI style scheme.
- **zero_background**: Flag indicating whether a background value of zero should be used.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

**Description**

Performs a fast approximate Gaussian filter on an image.

**Usage**

```r
wbt_fast_almost_gaussian_filter(
    input,  # Input raster D8 pointer file.
    output,  # Output raster file.
    sigma = 1.8,  # Standard deviation.
    wd = NULL,  # Working directory.
    verbose_mode = FALSE,  # Sets verbose mode.
    compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools.
    command_only = FALSE  # Return command rather than running tool.
)
```
Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **sigma**: Standard deviation distance in pixels.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_fd8_flow_accumulation**

*Fd8 flow accumulation*

Description

Calculates an FD8 flow accumulation raster from an input DEM.

Usage

```r
wbt_fd8_flow_accumulation(
  dem,
  output,
  out_type = "specific contributing area",
  exponent = 1.1,
  threshold = NULL,
  log = FALSE,
  clip = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
Arguments

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **out_type**: Output type; one of 'cells', 'specific contributing area' (default), and 'catchment area'.
- **exponent**: Optional exponent parameter; default is 1.1.
- **threshold**: Optional convergence threshold parameter, in grid cells; default is infinity.
- **log**: Optional flag to request the output be log-transformed.
- **clip**: Optional flag to request clipping the display max by 1 percent.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Calculates an FD8 flow pointer raster from an input DEM.

Usage

```r
wbt_fd8_pointer(
  dem, 
  output, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)
```
wbt_feature_preserving_smoothing

Arguments

- **dem**
  Input raster DEM file.

- **output**
  Output raster file.

- **wd**
  Changes the working directory.

- **verbose_mode**
  Sets verbose mode. If verbose mode is **FALSE**, tools will not print output messages.

- **compress_rasters**
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

- **command_only**
  Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

*wbt_feature_preserving_smoothing*

*Feature preserving smoothing*

Description

Reduces short-scale variation in an input DEM using a modified Sun et al. (2007) algorithm.

Usage

```r
wbt_feature_preserving_smoothing(
  dem,
  output,
  filter = 11,
  norm_diff = 15,
  num_iter = 3,
  max_diff = 0.5,
  zfactor = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **dem**
  Input raster DEM file.

- **output**
  Output raster file.

- **filter**
  Size of the filter kernel.
**norm_diff**  Maximum difference in normal vectors, in degrees.

**num_iter**  Number of iterations.

**max_diff**  Maximum allowable absolute elevation change (optional).

**zfactor**  Optional multiplier for when the vertical and horizontal units are not the same.

**wd**  Changes the working directory.

**verbose_mode**  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

**compress_rasters**  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

**command_only**  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_fetch_analysis**

**Description**

Performs an analysis of fetch or upwind distance to an obstacle.

**Usage**

```r
wbt_fetch_analysis(
  dem,
  output,
  azimuth = 0,
  hgt_inc = 0.05,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **dem**  Input raster DEM file.
- **output**  Output raster file.
- **azimuth**  Wind azimuth in degrees in degrees.
- **hgt_inc**  Height increment value.
- **wd**  Changes the working directory.
verbose_mode

Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters

Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only

Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_fill_burn

Fill burn

Description

Burns streams into a DEM using the FillBurn (Saunders, 1999) method.

Usage

wbt_fill_burn(
  dem,
  streams,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

dem
Input raster DEM file.

streams
Input vector streams file.

output
Output raster file.

wd
Changes the working directory.

verbose_mode
Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
**wbt_fill_depressions**  
*Fill depressions*

**Description**

Fills all of the depressions in a DEM. Depression breaching should be preferred in most cases.

**Usage**

```r
wbt_fill_depressions(
  dem,
  output,
  fix_flats = TRUE,
  flat_increment = NULL,
  max_depth = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **fix_flats**: Optional flag indicating whether flat areas should have a small gradient applied.
- **flat_increment**: Optional elevation increment applied to flat areas.
- **max_depth**: Optional maximum depression depth to fill.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
Description

Fills all of the depressions in a DEM using the Planchon and Darboux (2002) method.

Usage

```r
wbt_fill_depressions_planchon_and_darboux(
  dem,
  output,
  fix_flats = TRUE,
  flat_increment = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

dem Input raster DEM file.
output Output raster file.
fix_flats Optional flag indicating whether flat areas should have a small gradient applied.
flat_increment Optional elevation increment applied to flat areas.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**wbt_fill_depressions_wang_and_liu**

*Fill depressions wang and liu*

---

**Description**

Fills all of the depressions in a DEM using the Wang and Liu (2006) method. Depression breaching should be preferred in most cases.

**Usage**

```r
wbt_fill_depressions_wang_and_liu(
  dem,
  output,
  fix_flats = TRUE,
  flat_increment = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `dem` Input raster DEM file.
- `output` Output raster file.
- `fix_flats` Optional flag indicating whether flat areas should have a small gradient applied.
- `flat_increment` Optional elevation increment applied to flat areas.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**Description**

Fills NoData holes in a DEM.

**Usage**

```r
wbt_fill_missing_data(
  input,  # Input raster file.
  output,  # Output raster file.
  filter = 11,  # Filter size (cells).
  weight = 2,  # IDW weight value.
  no_edges = TRUE,  # Optional flag indicating whether to exclude NoData cells in edge regions.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE  # Return command that would be executed by system() rather than running tool.
)
```

**Arguments**

- `output`: Output raster file.
- `filter`: Filter size (cells).
- `weight`: IDW weight value.
- `no_edges`: Optional flag indicating whether to exclude NoData cells in edge regions.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.
wbt_fill_single_cell_pits

*Fill single cell pits*

**Description**

Raises pit cells to the elevation of their lowest neighbour.

**Usage**

```r
wbt_fill_single_cell_pits(
  dem,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dem</td>
<td>Input raster DEM file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.

---

wbt_filter_lidar_classes

*Filter lidar classes*

**Description**

Removes points in a LAS file with certain specified class values.
Usage

```r
wbt_filter_lidar_classes(
  input,
  output,
  exclude_cls = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input LiDAR file.
- **output**: Output LiDAR file.
- **exclude_cls**: Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, `exclude_cls='3,4,5,6,7,18'`.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_filter_lidar_scan_angles**

*Filter lidar scan angles*

Description

Removes points in a LAS file with scan angles greater than a threshold.

Usage

```r
wbt_filter_lidar_scan_angles(
  input,
  output,
  threshold,
  wd = NULL,
  verbose_mode = FALSE,
  ```
Arguments

- **input**: Input LiDAR file.
- **output**: Output LiDAR file.
- **threshold**: Scan angle threshold.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Removes small-area features from a raster.

Usage

```r
wbt_filter_raster_features_by_area(  
  input,  
  output,  
  threshold,  
  background = "zero",  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```
wbt_find_flightline_edge_points

Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **threshold**: Remove features with fewer grid cells than this threshold value.
- **background**: Background value.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

**Usage**

```r
wbt_find_flightline_edge_points(input, output, wd = NULL, verbose_mode = FALSE, compress_rasters = FALSE, command_only = FALSE)
```

**Arguments**

- **input**: Input LiDAR file.
- **output**: Output file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
wbt_find_lowest_or_highest_points

compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

wbt_find_lowest_or_highest_points
Find lowest or highest points

Description
Locates the lowest and/or highest valued cells in a raster.

Usage
wbt_find_lowest_or_highest_points(
  input, 
  output, 
  out_type = "lowest", 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE 
)

Arguments
input Input raster file.
output Output vector points file.
out_type Output type; one of 'area' (default) and 'volume'.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.
wbt_find_main_stem  Find main stem

**Description**
Finds the main stem, based on stream lengths, of each stream network.

**Usage**

```r
wbt_find_main_stem(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**
- **d8_pntr**: Input raster D8 pointer file.
- **streams**: Input raster streams file.
- **output**: Output raster file.
- **esri_pntr**: D8 pointer uses the ESRI style scheme.
- **zero_background**: Flag indicating whether a background value of zero should be used.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**
Returns the tool text outputs.
wbt_find_no_flow_cells

Find no flow cells

Description

Finds grid cells with no downslope neighbours.

Usage

wbt_find_no_flow_cells(
    dem, output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

dem            Input raster DEM file.
output         Output raster file.
wd             Changes the working directory.
verbose_mode   Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only   Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_find_parallel_flow

Find parallel flow

Description

Finds areas of parallel flow in D8 flow direction rasters.
wbt_find_patch_or_class_edge_cells

Find patch or class edge cells

Usage

wbt_find_patch_or_class_edge_cells(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

d8_pntr  Input D8 pointer raster file.
streams  Input raster streams file.
output   Output raster file.
wd       Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Finds all cells located on the edge of patch or class features.

Usage

wbt_find_patch_or_class_edge_cells(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input raster file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>

Value

Returns the tool text outputs.

### Description

Identifies potential ridge and peak grid cells.

### Usage

```r
wbt_find_ridges(
  dem,
  output,
  line_thin = TRUE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dem</td>
<td>Input raster DEM file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>line_thin</td>
<td>Optional flag indicating whether post-processing line-thinning should be performed.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>
wbt_fix_dangling_arcs

Description

This tool fixes undershot and overshot arcs, two common topological errors, in an input vector lines file.

Usage

wbt_fix_dangling_arcs(
    input,          
    output,         
    dist = "",      
    wd = NULL,      
    verbose_mode = FALSE,  
    compress_rasters = FALSE, 
    command_only = FALSE
)

Arguments

input          Name of the input lines vector file.
output         Name of the output lines vector file.
dist           Snap distance, in xy units (metres).
wd             Changes the working directory.
verbose_mode   Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only   Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
Flatten lakes

Description

Flattens lake polygons in a raster DEM.

Usage

```r
wbt_flatten_lakes(
  dem, 
  lakes, 
  output, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE 
)
```

Arguments

- `dem` Input raster DEM file.
- `lakes` Input lakes vector polygons file.
- `output` Output raster file.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**wbt_flightline_overlap**

*Flightline overlap*

**Description**

Reads a LiDAR (LAS) point file and outputs a raster containing the number of overlapping flight lines in each grid cell.

**Usage**

```r
wbt_flightline_overlap(
  input,
  output = NULL,
  resolution = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input LiDAR file.
- **output**: Output file.
- **resolution**: Output raster’s grid resolution.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**Description**

Reflects an image in the vertical or horizontal axis.

**Usage**

```r
wbt_flip_image(
  input,
  output,
  direction = "vertical",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **direction**: Direction of reflection; options include 'v' (vertical), 'h' (horizontal), and 'b' (both).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
### wbt_flood_order

**Flood order**

**Description**

Assigns each DEM grid cell its order in the sequence of inundations that are encountered during a search starting from the edges, moving inward at increasing elevations.

**Usage**

```r
wbt_flood_order(
  dem,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `dem` Input raster DEM file.
- `output` Output raster file.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

### wbt_floor

**Floor**

**Description**

Returns the largest (closest to positive infinity) value that is less than or equal to the values in a raster.
Usage

```r
wbt_floor(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_flow_accumulation_full_workflow**

*Flow accumulation full workflow*

Description

Resolves all of the depressions in a DEM, outputting a breached DEM, an aspect-aligned non-divergent flow pointer, and a flow accumulation raster.

Usage

```r
wbt_flow_accumulation_full_workflow(
  dem,
  out_dem,
  out_pntr,
  out_accum,
  out_type = "Specific Contributing Area",
  log = FALSE,
  clip = FALSE,
)```
Arguments

dem Input raster DEM file.
out_dem Output raster DEM file.
out_pntr Output raster flow pointer file.
out_accum Output raster flow accumulation file.
out_type Output type; one of ‘cells’, ‘sca’ (default), and ‘ca’.
log Optional flag to request the output be log-transformed.
clip Optional flag to request clipping the display max by 1 percent.
esri_pntr D8 pointer uses the ESRI style scheme.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Calculates the local maximum absolute difference in downslope flowpath length, useful in mapping drainage divides and ridges.

Usage

wbt_flow_length_diff(
d8_pntr, output, esri_pntr = FALSE, wd = NULL,
Arguments

d8_pntr Input D8 pointer raster file.
output Output raster file.
esri_pntr D8 pointer uses the ESRI style scheme.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Performs a gamma correction on an input image.

Usage

```r
wbt_gamma_correction(
  input,
  output,
  gamma = 0.5,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
### Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input raster file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>gamma</td>
<td>Gamma value.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output</td>
</tr>
<tr>
<td></td>
<td>messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression</td>
</tr>
<tr>
<td></td>
<td>for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>

### Value

Returns the tool text outputs.

---

**wbt_gaussian_contrast_stretch**

*Gaussian contrast stretch*

### Description

Performs a Gaussian contrast stretch on input images.

### Usage

```r
wbt_gaussian_contrast_stretch(
  input,
  output,
  num_tones = 256,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input raster file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>num_tones</td>
<td>Number of tones in the output image.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output</td>
</tr>
<tr>
<td></td>
<td>messages.</td>
</tr>
</tbody>
</table>
wbt_gaussian_curvature

compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_gaussian_curvature

*Gaussian curvature*

Description

Calculates a mean curvature raster from an input DEM.

Usage

```r
wbt_gaussian_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dem</td>
<td>Input raster DEM file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>log</td>
<td>Display output values using a log-scale.</td>
</tr>
<tr>
<td>zfactor</td>
<td>Optional multiplier for when the vertical and horizontal units are not the same.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>

Value

Returns the tool text outputs.
wbt_gaussian_filter  Gaussian filter

Description

Performs a Gaussian filter on an image.

Usage

wbt_gaussian_filter(
    input,
    output,
    sigma = 0.75,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **sigma**: Standard deviation distance in pixels.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
wbt_gaussian_scale_space

Gaussian scale space

Description

This tool uses the fast Gaussian approximation algorithm to produce scaled land-surface parameter measurements from an input DEM.

Usage

wbt_gaussian_scale_space(
  dem,
  output,
  output_zscore,
  output_scale,
  points = NULL,
  sigma = 0.5,
  step = 0.5,
  num_steps = 10,
  lsp = "Slope",
  z_factor = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

dem       Name of the input DEM raster file.
output    Name of the output land-surface parameter raster file.
output_zscore    Name of the output z-score raster file.
output_scale    Name of the output scale raster file.
points     Name of the input vector points shapefile.
sigma      Initial sigma value (cells).
step       Step size as any positive non-zero integer.
num_steps  Number of steps.
lsp        Output land-surface parameter; one of 'AnisotropyLTP', 'Aspect', 'DiffMeanElev', 'Eastness', 'Elevation', 'Hillshade', 'MeanCurvature', 'Northness', 'PlanCurvature', 'ProfileCurvature', 'Ruggedness', 'Slope', 'TanCurvature', 'TotalCurvature'.
z_factor   Optional multiplier for when the vertical and horizontal units are not the same.
wd         Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

---

wbt_generalize_classified_raster

*Generalize classified raster*

Description
Generalizes a raster containing class or object features by removing small features.

Usage
```r
wbt_generalize_classified_raster(
  input,
  output,
  min_size = 4,
  method = "longest",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments
- **input**: Name of the input raster image file.
- **output**: Name of the output raster file.
- **min_size**: Minimum feature size, in grid cells.
- **method**: Grouping method; one of 'longest' (default), 'largest', and 'nearest'.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.
Value

Returns the tool text outputs.

---

**wbt_generalize_with_similarity**

*Generalize with similarity*

**Description**

Generalizes a raster containing class or object features by removing small features using similarity criteria of neighbouring features.

**Usage**

```r
wbt_generalize_with_similarity(
  input,
  similarity,
  output,
  min_size = 4,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Name of the input raster image file.
- **similarity**: Names of the input similarity images.
- **output**: Name of the output raster file.
- **min_size**: Minimum feature size, in grid cells.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_generating_function

Generating function

Description

This tool calculates generating function from an input DEM.

Usage

```r
wbt_generating_function(
  dem,
  output,
  log = FALSE,
  zfactor = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

dem  Name of the input raster DEM file.
output  Name of the output raster image file.
log  Display output values using a log-scale.
zfactor  Z conversion factor.
w  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**wbt_geomorphons**  
*Geomorphons*

**Description**  
Computes geomorphon patterns.

**Usage**

```r
wbt_geomorphons(
  dem,
  output,
  search = 50,
  threshold = 0,
  tdist = 0,
  forms = TRUE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **dem**  
  Input raster DEM file.
- **output**  
  Output raster file.
- **search**  
  Look up distance.
- **threshold**  
  Flatness threshold for the classification function (in degrees).
- **tdist**  
  Distance (in cells) to begin reducing the flatness threshold to avoid problems with pseudo-flat lines-of-sight.
- **forms**  
  Classify geomorphons into 10 common land morphologies, else, output ternary code.
- **wd**  
  Changes the working directory.
- **verbose_mode**  
  Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**  
  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_greater_than

Greater than

Description

Performs a greater-than comparison operation on two rasters or a raster and a constant value.

Usage

```r
wbt_greater_than(
  input1,
  input2,
  output,
  incl_equals = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input1**: Input raster file or constant value.
- **input2**: Input raster file or constant value.
- **output**: Output raster file.
- **incl_equals**: Perform a greater-than-or-equal-to operation.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_hack_stream_order  Hack stream order

Description

Assigns the Hack stream order to each tributary in a stream network.

Usage

wbt_hack_stream_order(
    d8_pntr,
    streams,
    output,
    esri_pntr = FALSE,
    zero_background = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

- **d8_pntr**: Input raster D8 pointer file.
- **streams**: Input raster streams file.
- **output**: Output raster file.
- **esri_pntr**: D8 pointer uses the ESRI style scheme.
- **zero_background**: Flag indicating whether a background value of zero should be used.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_height_above_ground

*Height above ground*

**Description**

Normalizes a LiDAR point cloud, providing the height above the nearest ground-classified point.

**Usage**

```r
wbt_height_above_ground(
  input,
  output = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input`: Input LiDAR file (including extension).
- `output`: Output raster file (including extension).
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_help**

*Help description for WhiteboxTools*

**Description**

Help description for WhiteboxTools

**Usage**

`wbt_help()`
wbt_highest_position

Value

Returns the help description for WhiteboxTools as an R character vector.

Examples

## Not run:
wbt_help()

## End(Not run)

wbt_highest_position  Highest position

Description

Identifies the stack position of the maximum value within a raster stack on a cell-by-cell basis.

Usage

wbt_highest_position(  
    inputs,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)

Arguments

inputs        Input raster files.
output        Output raster file.
wd            Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
wbt_high_pass_filter  

High pass filter

Description

Performs a high-pass filter on an input image.

Usage

wbt_high_pass_filter(
  input,
  output,
  filterx = 11,
  filtery = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input  Input raster file.
output  Output raster file.
filterx  Size of the filter kernel in the x-direction.
filtery  Size of the filter kernel in the y-direction.
wpector  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
**wbt_high_pass_median_filter**

*High pass median filter*

**Description**

Performs a high pass median filter on an input image.

**Usage**

```r
call = wbt_high_pass_median_filter(
  input, output,
  filterx = 11,
  filtety = 11,
  sig_digits = 2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `output`: Output raster file.
- `filterx`: Size of the filter kernel in the x-direction.
- `filtety`: Size of the filter kernel in the y-direction.
- `sig_digits`: Number of significant digits.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_hillshade**

**Hillshade**

**Description**
Calculates a hillshade raster from an input DEM.

**Usage**

```r
wbt_hillshade(
  dem,
  output,
  azimuth = 315,
  altitude = 30,
  zfactor = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `dem` Input raster DEM file.
- `output` Output raster file.
- `azimuth` Illumination source azimuth in degrees.
- `altitude` Illumination source altitude in degrees.
- `zfactor` Optional multiplier for when the vertical and horizontal units are not the same.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**Description**

Identifies the individual hillslopes draining to each link in a stream network.

**Usage**

```r
wbt_hillslopes(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>d8_pntr</td>
<td>Input raster D8 pointer file.</td>
</tr>
<tr>
<td>streams</td>
<td>Input raster streams file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>esri_pntr</td>
<td>D8 pointer uses the ESRI style scheme.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.
wbt_histogram_equalization

Histogram equalization

Description

Performs a histogram equalization contrast enhancement on an image.

Usage

wbt_histogram_equalization(
    input, output,
    num_tones = 256,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input       Input raster file.
output      Output raster file.
num_tones   Number of tones in the output image.
wd          Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
**wbt_histogram_matching**

*Histogram matching*

**Description**

Alters the statistical distribution of a raster image matching it to a specified PDF.

**Usage**

```r
wbt_histogram_matching(
  input, 
  histo_file, 
  output, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input raster file.</td>
</tr>
<tr>
<td>histo_file</td>
<td>Input reference probability distribution function (pdf) text file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.
Description

This tool alters the cumulative distribution function of a raster image to that of another image.

Usage

```r
wbt_histogram_matching_two_images(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input1**: Input raster file to modify.
- **input2**: Input reference raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**wbt_hole_proportion**

**Hole proportion**

**Description**

Calculates the proportion of the total area of a polygon’s holes relative to the area of the polygon’s hull.

**Usage**

```r
wbt_hole_proportion(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input vector polygon file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_horizontal_excess_curvature**

**Horizontal excess curvature**

**Description**

This tool calculates horizontal excess curvature from an input DEM.
Usage

```r
wbt_horizontal_excess_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

dem Name of the input raster DEM file.
output Name of the output raster image file.
log Display output values using a log-scale.
zfactor Z conversion factor.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

wbt_horizon_angle Horizon angle

Description

Calculates horizon angle (maximum upwind slope) for each grid cell in an input DEM.

Usage

```r
wbt_horizon_angle(
  dem,
  output,
  azimuth = 0,
  max_dist = 100,
  wd = NULL,
)```
Arguments

dem Input raster DEM file.
output Output raster file.
azimuth Azimuth, in degrees.
max_dist Optional maximum search distance (unspecified if none; in xy units). Minimum value is 5 x cell size.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_horton_stream_order**

*Horton stream order*

**Description**

Assigns the Horton stream order to each tributary in a stream network.

**Usage**

```r
wbt_horton_stream_order(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
wbt_hydrologic_connectivity

Arguments

- streams: Input raster streams file.
- output: Output raster file.
- esri_pntr: D8 pointer uses the ESRI style scheme.
- zero_background: Flag indicating whether a background value of zero should be used.
- wd: Changes the working directory.
- verbose_mode: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- compress_rasters: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- command_only: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

wbt_hydrologic_connectivity

Hydrologic connectivity

Description

This tool evaluates hydrologic connectivity within a DEM.

Usage

```r
wbt_hydrologic_connectivity(
  dem,
  output1,
  output2,
  exponent = 1,
  threshold = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
wbt_hypsometrically_tinted_hillshade

Arguments

dem Name of the input DEM raster file; must be depressionless.
output1 Name of the output downslope unsaturated length (DUL) file.
output2 Name of the output upslope disconnected saturated area (UDSA) file.
exponent Optional exponent parameter; default is 1.0.
threshold Optional convergence threshold parameter, in grid cells; default is infinity.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_hypsometrically_tinted_hillshade

Hypsometrically tinted hillshade

Description

Creates an colour shaded relief image from an input DEM.

Usage

wbt_hypsometrically_tinted_hillshade(
  dem,
  output,
  altitude = 45,
  hs_weight = 0.5,
  brightness = 0.5,
  atmospheric = 0,
  palette = "atlas",
  reverse = FALSE,
  zfactor = NULL,
  full_mode = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
Arguments

- output: Output raster file.
- altitude: Illumination source altitude in degrees.
- hs_weight: Weight given to hillshade relative to relief (0.0-1.0).
- brightness: Brightness factor (0.0-1.0).
- atmospheric: Atmospheric effects weight (0.0-1.0).
- palette: Options include 'atlas', 'high_relief', 'arid', 'soft', 'muted', 'purple', 'viridi', 'gn_yl', 'pi_y_g', 'bl_yl_rd', and 'deep'.
- reverse: Optional flag indicating whether to use reverse the palette.
- zfactor: Optional flag indicating whether to use full 360-degrees of illumination sources.
- wd: Changes the working directory.
- verbose_mode: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- compress_rasters: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- command_only: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Calculates a hypsometric curve for one or more DEMs.

Usage

```r
wbt_hypsometric_analysis(inputs, output, watershed = NULL, wd = NULL, verbose_mode = FALSE, compress_rasters = FALSE, command_only = FALSE)
```
Arguments

inputs  Input DEM files.
output  Output HTML file (default name will be based on input file if unspecified).
watershed  Input watershed files (optional).
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

Idw interpolation

Description

Interpolates vector points into a raster surface using an inverse-distance weighted scheme.

Usage

```r
wbt_idw_interpolation(
  input,
  field,
  output,
  use_z = FALSE,
  weight = 2,
  radius = NULL,
  min_points = NULL,
  cell_size = NULL,
  base = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
Arguments

- **input**: Input vector Points file.
- **field**: Input field name in attribute table.
- **output**: Output raster file.
- **use_z**: Use z-coordinate instead of field?.
- **weight**: IDW weight value.
- **radius**: Search Radius in map units.
- **min_points**: Minimum number of points.
- **cell_size**: Optionally specified cell size of output raster. Not used when base raster is specified.
- **base**: Optionally specified input base raster file. Not used when a cell size is specified.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_ihs_to_rgb**  
*Ihs to rgb*

Description

Converts intensity, hue, and saturation (IHS) images into red, green, and blue (RGB) images.

Usage

```r
wbt_ihs_to_rgb(  
  intensity,  
  hue,  
  saturation,  
  red = NULL,  
  green = NULL,  
  blue = NULL,  
  output = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)
```
wbt_image_autocorrelation

Arguments

- **intensity**: Input intensity file.
- **hue**: Input hue file.
- **saturation**: Input saturation file.
- **red**: Output red band file. Optionally specified if colour-composite not specified.
- **green**: Output green band file. Optionally specified if colour-composite not specified.
- **blue**: Output blue band file. Optionally specified if colour-composite not specified.
- **output**: Output colour-composite file. Only used if individual bands are not specified.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_image_autocorrelation

*Image autocorrelation*

Description

Performs Moran’s I analysis on two or more input images.

Usage

```r
wbt_image_autocorrelation(
    inputs, output,
    contiguity = "Rook",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```
Arguments

inputs  Input raster files.
output  Output HTML file (default name will be based on input file if unspecified).
contiguity  Contiguity type.
wdd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_image_correlation  Image correlation

Description

Performs image correlation on two or more input images.

Usage

wbt_image_correlation(
  inputs,
  output = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

inputs  Input raster files.
output  Output HTML file (default name will be based on input file if unspecified).
wdd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.
Value
Returns the tool text outputs.

Usage

```r
wbt_image_correlation_neighbourhood_analysis(
  input1,
  input2,
  output1,
  output2,
  filter = 11,
  stat = "pearson",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input1**: Input raster file.
- **input2**: Input raster file.
- **output1**: Output correlation (r-value or rho) raster file.
- **output2**: Output significance (p-value) raster file.
- **filter**: Size of the filter kernel.
- **stat**: Correlation type; one of 'pearson' (default) and 'spearman'.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value
Returns the tool text outputs.
**Description**

Performs image regression analysis on two input images.

**Usage**

```r
wbt_image_regression(
  input1,
  input2,
  output,
  out_residuals = NULL,
  standardize = FALSE,
  scattergram = FALSE,
  num_samples = 1000,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input1`: Input raster file (independent variable, X).
- `input2`: Input raster file (dependent variable, Y).
- `output`: Output HTML file for regression summary report.
- `standardize`: Optional flag indicating whether to standardize the residuals map.
- `scattergram`: Optional flag indicating whether to output a scattergram.
- `num_samples`: Number of samples used to create scattergram.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_image_segmentation

Image segmentation

Description

Performs a region-growing based segmentation on a set of multi-spectral images.

Usage

```r
wbt_image_segmentation(
  inputs,
  output,
  threshold = 0.5,
  steps = 10,
  min_area = 4,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **inputs**: Names of the input band images.
- **output**: Name of the output raster file.
- **threshold**: Distance threshold, in z-scores.
- **steps**: Number of steps.
- **min_area**: Minimum object area, in grid cells (1-8).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
Description

This tool creates an image slider from two input images.

Usage

```r
wbt_image_slider(
  input1,
  input2,
  output,
  palette1 = "grey",
  reverse1 = FALSE,
  label1 = "",
  palette2 = "grey",
  reverse2 = FALSE,
  label2 = "",
  height = 600,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input1**: Name of the left input image file.
- **input2**: Name of the right input image file.
- **output**: Name of the output HTML file (*.html).
- **palette1**: Left image palette; options are 'grey', 'atlas', 'high_relief', 'arid', 'soft', 'muted', 'purple', 'viridi', 'gn_yl', 'pi_y_g', 'bl_yl_rd', 'deep', and 'rgb'.
- **reverse1**: Reverse left image palette?.
- **label1**: Left image label (leave blank for none).
- **palette2**: Right image palette; options are 'grey', 'atlas', 'high_relief', 'arid', 'soft', 'muted', 'purple', 'viridi', 'gn_yl', 'pi_y_g', 'bl_yl_rd', 'deep', and 'rgb'.
- **reverse2**: Reverse right image palette?.
- **label2**: Right image label (leave blank for none).
- **height**: Image height, in pixels.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
wbt_image_stack_profile

compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

wbt_image_stack_profile

Image stack profile

Description
Plots an image stack profile (i.e. signature) for a set of points and multispectral images.

Usage
wbt_image_stack_profile(
  inputs,
  points,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments
inputs
Input multispectral image files.
points
Input vector points file.
output
Output HTML file.
wd
Changes the working directory.
verbose_mode
Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only
Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.
**wbt_impoundment_size_index**

*Impoundment size index*

**Description**

Calculates the impoundment size resulting from damming a DEM.

**Usage**

```r
wbt_impoundment_size_index(
  dem,
  damlength,
  out_mean = NULL,
  out_max = NULL,
  out_volume = NULL,
  out_area = NULL,
  out_dam_height = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **dem**: Input raster DEM file.
- **damlength**: Maximum length of the dam.
- **out_mean**: Output mean flooded depth file.
- **out_max**: Output maximum flooded depth file.
- **out_volume**: Output flooded volume file.
- **out_area**: Output flooded area file.
- **out_dam_height**: Output dam height file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_increment

Increment

Description

Increases the values of each grid cell in an input raster by 1.0. (see also InPlaceAdd).

Usage

wbt_increment(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input Input raster file.
output Output raster file.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_init

Initialize WhiteboxTools
Description

wbt_init(): Check if a suitable WhiteboxTools executable is present. Search default path in package directory or set it manually with exe_path.

wbt_options(): Get/set package options

- whitebox.exe_path - character. Path to executable file. The default value is the package installation directory, subdirectory "WBT", followed by whitebox_tools.exe or whitebox_tools. Set the whitebox.exe_path option using wbt_init() exe_path argument

- whitebox.wd - character. Path to WhiteboxTools working directory. Used as --wd argument for tools that support it when wd is not specified elsewhere.

- whitebox.verbose - logical. Should standard output from calls to executable be cat() output for readability? Default is result of interactive(). Individual tools may have verbose_mode setting that produce only single-line output when FALSE. These argument values are left as the defaults defined in the package documentation for that function. When whitebox.verbose=FALSE no output is produced. Set the value of whitebox.verbose with wbt_verbose() verbose argument.

- whitebox.compress_rasters - logical. Should raster output from WhiteboxTools be compressed? Default: FALSE. Set the value of whitebox.compress_rasters with wbt_compress_rasters() compress_rasters argument.

- whitebox.max_procs - integer. Maximum number of processes for tools that run in parallel or partially parallelize. Default: -1 uses all of the available cores.

wbt_exe_path(): Get the file path of the WhiteboxTools executable.

wbt_wd(): Get or set the WhiteboxTools working directory. Default: "" (unset) is your R working directory if no other options are set.

wbt_verbose(): Check verbose options for WhiteboxTools

wbt_compress_rasters(): Check raster compression option for WhiteboxTools. Default: FALSE

wbt_max_procs(): Check maximum number of processes for for tools that run in parallel or partially parallelize. Default: -1 uses all of the available cores.

Usage

wbt_init(exe_path = wbt_exe_path(shell_quote = FALSE), ...)

wbt_options(
  exe_path = NULL,
  wd = NULL,
  verbose = NULL,
  compress_rasters = NULL,
  max_procs = NULL
)

wbt_exe_path(exe_path = NULL, shell_quote = TRUE)

wbt_default_path()
```r
wbt_init

wbt_wd(wd = NULL)

wbt_verbose(verbos = NULL)

wbt_compress_rasters(compress_rasters = NULL)

wbt_max_procs(max_procs = NULL)

Arguments

exe_path     Optional: User-supplied path to WhiteboxTools executable. Default: NULL
...
additional arguments to wbt_options()
wd           character; Default: NULL; if set the package option whitebox.wd is set specified
             path (if directory exists)
verbose      Default: NULL; if logical, set the package option whitebox.verbose to specified
             value
compress_rasters Default: NULL; if logical, set the package option whitebox.compress_rasters
             to specified value
max_procs    Default: NULL; if integer, set the package option whitebox.max_procs to spec-
             ified value
shell_quote  Return shQuote() result?

Details

wbt_exe_path(): Checks system environment variable R_WHITEBOX_EXE_PATH and package op-
              tion whitebox.exe_path. Set your desired path with either Sys.setenv(R_WHITEBOX_EXE_PATH =
              "C:/path/to/whitebox_tools.exe") or options(whitebox.exe_path = "C:/path/to/whitebox_tools.exe").
              The default, backwards-compatible path is returned by wbt_default_path()

wbt_wd(): Before you set the working directory in a session the default output will be in your
          current R working directory unless otherwise specified. You can change working directory at any
          time by setting the wd argument to wbt_wd() and running a tool. Note that once you have set a
          working directory, the directory needs to be set somewhere to "replace" the old value; just dropping
          the flag will not change the working directory back to the R working directory. To "unset" the option
          in the R package you can use wbt_wd(""") which is equivalent to wbt_wd(getwd())

Value

wbt_init(): logical; TRUE if binary file is found at exe_path

wbt_options(): an invisible list containing current whitebox.exe_path, whitebox.verbose, 
              whitebox.compress_rasters, and whitebox.max_procs options

Returns the file path of WhiteboxTools executable.

wbt_wd(): character; when working directory is unset, will not add --wd= arguments to calls and
          should be the same as using getwd(). See Details.

wbt_verbose(): logical; defaults to result of interactive()

wbt_compress_rasters(): logical; defaults to FALSE

wbt_max_procs(): integer; defaults to -1
```
See Also

install_whitebox() whitebox

Examples

## Not run:
## wbt_init():

# set path to binary as an argument
# wbt_init(exe_path = "not/a/valid/path/whitebox_tools.exe")

## End(Not run)
## Not run:

## wbt_options():

# set multiple options (e.g. exe_path and verbose) with wbt_options()
wbt_options(exe_path = "not/a/valid/path/whitebox_tools.exe", verbose = TRUE)

## End(Not run)
## Not run:

wbt_exe_path()

## End(Not run)
## Not run:

## wbt_wd():

# set WBT working directory to R working directory
wbt_wd(wd = getwd())

## End(Not run)
## Not run:

## wbt_verbose():

wbt_verbose(verbose = TRUE)

## End(Not run)
## Not run:

## wbt_compress_rasters():

wbt_compress_rasters(compress_rasters = TRUE)

## End(Not run)
## Not run:

## wbt_max_procs():

wbt_max_procs(max_procs = 2)
## Insert dams

Calculates the impoundment size resulting from damming a DEM.

### Usage

```r
wbt_insert_dams(
  dem, 
  dam_pts, 
  output, 
  damlength, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE 
)
```

### Arguments

- `dem` Input raster DEM file.
- `dam_pts` Input vector dam points file.
- `output` Output file.
- `damlength` Maximum length of the dam.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

### Value

Returns the tool text outputs.
**wbt_install**  
*Download and Install WhiteboxTools*

**Description**

This function downloads the WhiteboxTools binary if needed. Pre-compiled binaries are only available for download for 64-bit Linux (Ubuntu 20.04), Windows and Mac OS (Intel) platforms. If you need WhiteboxTools for another platform follow the instructions here: [https://github.com/jblindsay/whitebox-tools](https://github.com/jblindsay/whitebox-tools)

**Usage**

```r
wbt_install(pkg_dir = find.package("whitebox"), force = FALSE)
install_whitebox(pkg_dir = find.package("whitebox"), force = FALSE)
```

**Arguments**

- `pkg_dir`: default install path is to whitebox package "WBT" folder
- `force`: logical. Default FALSE. Force install?

**Value**

Prints out the location of the WhiteboxTools binary, if found. NULL otherwise.

**Examples**

```r
## Not run:
install_whitebox()
## End(Not run)
```

---

**wbt_integer_division**  
*Integer division*

**Description**

Performs an integer division operation on two rasters or a raster and a constant value.
Usage

wbt_integer_division(
    input1,
    input2,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input1 Input raster file or constant value.
input2 Input raster file or constant value.
output Output raster file.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_integral_image Integral image

Description

Transforms an input image (summed area table) into its integral image equivalent.

Usage

wbt_integral_image(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is **FALSE**, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_intersect**  
**Intersect**

Description

Identifies the parts of features in common between two input vector layers.

Usage

```r
wbt_intersect(
  input,  
  overlay, 
  output, 
  snap = 0, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE 
)
```

Arguments

- **input**: Input vector file.
- **overlay**: Input overlay vector file.
- **output**: Output vector file.
- **snap**: Snap tolerance.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is **FALSE**, tools will not print output messages.
wbt_inverse_principal_component_analysis

Description

This tool performs an inverse principal component analysis on a series of input component images.

Usage

wbt_inverse_principal_component_analysis(  
    inputs,  
    report,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE
)

Arguments

inputs Name of the input PCA component images.
report Name of the PCA report file (*.html).
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
In place add

Description
Performs an in-place addition operation (input1 += input2).

Usage

```r
wbt_in_place_add(
  input1, 
  input2, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)
```

Arguments

- `input1`: Input raster file.
- `input2`: Input raster file or constant value.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

In place divide

Description
Performs an in-place division operation (input1 /= input2).
**wbt_in_place_multiply**

**Usage**

```r
wbt_in_place_multiply(
    input1,
    input2,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

- `input1` : Input raster file.
- `input2` : Input raster file or constant value.
- `wd` : Changes the working directory.
- `verbose_mode` : Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` : Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` : Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_in_place_multiply**  *In place multiply*

---

**Description**

Performs an in-place multiplication operation (input1 *= input2).

**Usage**

```r
wbt_in_place_multiply(
    input1,
    input2,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```
**Arguments**

- **input1**: Input raster file.
- **input2**: Input raster file or constant value.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_in_place_subtract  In place subtract**

**Description**

Performs an in-place subtraction operation (input1 -= input2).

**Usage**

```r
wbt_in_place_subtract(
  input1,
  input2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input1**: Input raster file.
- **input2**: Input raster file or constant value.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.
**Value**

Returns the tool text outputs.

---

### Description

Divides a landscape into nearly equal sized drainage basins (i.e. watersheds).

### Usage

```r
wbt_isobasins(
  dem,
  output,
  size,
  connections = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **size**: Target basin size, in grid cells.
- **connections**: Output upstream-downstream flow connections among basins?.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

### Value

Returns the tool text outputs.
wbt_is_no_data  Is no data

Description

Identifies NoData valued pixels in an image.

Usage

wbt_is_no_data(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input  Input raster file.
output  Output raster file.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_jenson_snap_pour_points  Jenson snap pour points

Description

Moves outlet points used to specify points of interest in a watershedding operation to the nearest stream cell.
wbt_join_tables

Usage

wbt_jenson_snap_pour_points(
    pour_pts,
    streams,
    output,
    snap_dist,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

pour_pts  Input vector pour points (outlet) file.
streams   Input raster streams file.
output    Output vector file.
snap_dist Maximum snap distance in map units.
wd        Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_join_tables  Join tables

Description

Merge a vector’s attribute table with another table based on a common field.

Usage

wbt_join_tables(
    input1,
    pkey,
    input2,
    fkey,
    import_field,
    )
wbt_kappa_index

Arguments

input1 Input primary vector file (i.e. the table to be modified).
pkey Primary key field.
input2 Input foreign vector file (i.e. source of data to be imported).
fkey Foreign key field.
import_field Imported field (all fields will be imported if not specified).
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Performs a kappa index of agreement (KIA) analysis on two categorical raster files.

Usage

wbt_kappa_index(
  input1, input2, output, wd = NULL, verbose_mode = FALSE, compress_rasters = FALSE, command_only = FALSE
)
**Arguments**

- **input1**: Input classification raster file.
- **input2**: Input reference raster file.
- **output**: Output HTML file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_knn_classification**

*Knn classification*

**Description**

Performs a supervised k-nearest neighbour classification using training site polygons/points and predictor rasters.

**Usage**

```r
wbt_knn_classification(
  inputs, 
  training, 
  field, 
  output, 
  scaling = "Normalize", 
  k = 5, 
  clip = TRUE, 
  test_proportion = 0.2, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE 
)
```
Arguments

- **inputs**: Names of the input predictor rasters.
- **training**: Name of the input training site polygons/points shapefile.
- **field**: Name of the attribute containing class name data.
- **output**: Name of the output raster file.
- **scaling**: Scaling method for predictors. Options include 'None', 'Normalize', and 'Standardize'.
- **k**: k-parameter, which determines the number of nearest neighbours used.
- **clip**: Perform training data clipping to remove outlier pixels?
- **test_proportion**: The proportion of the dataset to include in the test split; default is 0.2.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_knn_regression**  
Knn regression

Description

Performs a supervised k-nearest neighbour regression using training site points and predictor rasters.

Usage

```r
wbt_knn_regression(
  inputs, training, field, 
  scaling = "Normalize", output = NULL, k = 5, 
  weight = TRUE, test_proportion = 0.2, 
  wd = NULL, verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)
```
Arguments

inputs  Names of the input predictor rasters.
training Name of the input training site points Shapefile.
field Name of the attribute containing response variable name data.
scaling Scaling method for predictors. Options include 'None', 'Normalize', and 'Standardize'.
output Name of the output raster file.
k k-parameter, which determines the number of nearest neighbours used.
weight Use distance weighting?.
test_proportion The proportion of the dataset to include in the test split; default is 0.2.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Evaluates whether the values in a raster are normally distributed.

Usage

```r
wbt_ks_test_for_normality(
  input,
  output,
  num_samples = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
Arguments

input  Input raster file.
output  Output HTML file.
num_samples  Number of samples. Leave blank to use whole image.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_k_means_clustering

K means clustering

Description

Performs a k-means clustering operation on a multi-spectral dataset.

Usage

wbt_k_means_clustering(
  inputs,
  output,
  classes,
  out_html = NULL,
  max_iterations = 10,
  class_change = 2,
  initialize = "diagonal",
  min_class_size = 10,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
**wbt_k_nearest_mean_filter**

**Arguments**

- **inputs**: Input raster files.
- **output**: Output raster file.
- **classes**: Number of classes.
- **out_html**: Output HTML report file.
- **max_iterations**: Maximum number of iterations.
- **class_change**: Minimum percent of cells changed between iterations before completion.
- **initialize**: How to initialize cluster centres?
- **min_class_size**: Minimum class size, in pixels.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_k_nearest_mean_filter**

*K nearest mean filter*

---

**Description**

A k-nearest mean filter is a type of edge-preserving smoothing filter.

**Usage**

```r
wbt_k_nearest_mean_filter(
    input,
    output,
    filterx = 11,
    filtery = 11,
    k = 5,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```
Arguments

input  Input raster file.
output Output raster file.
filterx Size of the filter kernel in the x-direction.
filtery Size of the filter kernel in the y-direction.
k k-value in pixels; this is the number of nearest-valued neighbours to use.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

wbt_laplacian_filter Laplacian filter

Description

Performs a Laplacian filter on an image.

Usage

wbt_laplacian_filter(
    input, 
    output, 
    variant = "3x3(1)", 
    clip = 0, 
    wd = NULL, 
    verbose_mode = FALSE, 
    compress_rasters = FALSE, 
    command_only = FALSE 
)

Arguments

input  Input raster file.
output Output raster file.
variant Optional variant value. Options include 3x3(1), 3x3(2), 3x3(3), 3x3(4), 5x5(1), and 5x5(2) (default is 3x3(1)).
clip Optional amount to clip the distribution tails by, in percent.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value Returns the tool text outputs.

---

### Description

Performs a Laplacian-of-Gaussian (LoG) filter on an image.

### Usage

```r
wbt_laplacian_of_gaussian_filter(
  input,
  output,
  sigma = 0.75,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input raster file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>sigma</td>
<td>Standard deviation in pixels.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>
Value

Returns the tool text outputs.

---

**WBT_LAS_TO_ASCII**  
*Las to ascii*

---

**Description**

Converts one or more LAS files into ASCII text files.

**Usage**

```r
wbt_las_to_ascii(
  inputs,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **inputs**: Input LiDAR files.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_las_to_laz**

**Las to laz**

**Description**

This tool converts one or more LAS files into the LAZ format.

**Usage**

```r
wbt_las_to_laz(
  input,
  output = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Name of the input LAS files (leave blank to use all LAS files in WorkingDirectory).
- **output**: Output LAZ file (including extension).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_las_to_multipoint_shapefile**

**Las to multipoint shapefile**

**Description**

Converts one or more LAS files into MultipointZ vector Shapefiles. When the input parameter is not specified, the tool grids all LAS files contained within the working directory.
**Usage**

```r
wbt_las_to_multipoint_shapefile(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>input</code></td>
<td>Input LiDAR file.</td>
</tr>
<tr>
<td><code>wd</code></td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td><code>verbose_mode</code></td>
<td>Sets verbose mode. If verbose mode is <code>FALSE</code>, tools will not print output messages.</td>
</tr>
<tr>
<td><code>compress_rasters</code></td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td><code>command_only</code></td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.

**Description**

Converts one or more LAS files into a vector Shapefile of POINT ShapeType.

**Usage**

```r
wbt_las_to_shapefile(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
wbt_las_to_zlidar

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input LiDAR file.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>

Value

Returns the tool text outputs.

---

wbt_las_to_zlidar  Las to zlidar

Description

Converts one or more LAS files into the zlidar compressed LiDAR data format.

Usage

```r
wbt_las_to_zlidar(
  inputs = NULL,
  outdir = NULL,
  compress = "brotli",
  level = 5,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>inputs</td>
<td>Input LAS files.</td>
</tr>
<tr>
<td>outdir</td>
<td>Output directory into which zlidar files are created. If unspecified, it is assumed to be the same as the inputs.</td>
</tr>
<tr>
<td>compress</td>
<td>Compression method, including 'brotli' and 'deflate'.</td>
</tr>
<tr>
<td>level</td>
<td>Compression level (1-9).</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
</tbody>
</table>
compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

wbt_layer_footprint  Layer footprint

Description
Creates a vector polygon footprint of the area covered by a raster grid or vector layer.

Usage
wbt_layer_footprint(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments
input  Input raster or vector file.
output  Output vector polygon file.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.
**wbt_laz_to_las**  
Laz to las

**Description**

This tool converts one or more LAZ files into the LAS format.

**Usage**

```r
wbt_laz_to_las(  
  input,  
  output = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

- `input`: Name of the input LAZ files (leave blank to use all LAZ files in WorkingDirectory).
- `output`: Output LAS file (including extension).
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If `verbose_mode` is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_lee_sigma_filter**  
Lee sigma filter

**Description**

Performs a Lee (Sigma) smoothing filter on an image.
Usage

```r
wbt_lee_sigma_filter(
  input, output,
  filterx = 11,
  filtery = 11,
  sigma = 10,
  m = 5,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **filterx**: Size of the filter kernel in the x-direction.
- **filtery**: Size of the filter kernel in the y-direction.
- **sigma**: Sigma value should be related to the standard deviation of the distribution of image speckle noise.
- **m**: M-threshold value the minimum allowable number of pixels within the intensity range.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_length_of_upstream_channels**

*Length of upstream channels*

Description

Calculates the total length of channels upstream.
usage

```
wbt_length_of_upstream_channels(
    d8_pntr,
    streams,
    output,
    esri_pntr = FALSE,
    zero_background = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

Arguments

- `d8_pntr`  
  Input raster D8 pointer file.
- `streams`  
  Input raster streams file.
- `output`  
  Output raster file.
- `esri_pntr`  
  D8 pointer uses the ESRI style scheme.
- `zero_background`  
  Flag indicating whether a background value of zero should be used.
- `wd`  
  Changes the working directory.
- `verbose_mode`  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`  
  Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_less_than**  
*Less than*

Description

Performs a less-than comparison operation on two rasters or a raster and a constant value.
Usage

```r
wbt_less_than(
  input1,
  input2,
  output,
  incl_equals = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `input1`: Input raster file or constant value.
- `input2`: Input raster file or constant value.
- `output`: Output raster file.
- `incl_equals`: Perform a less-than-or-equal-to operation.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

License information for WhiteboxTools

```r
wbt_license()
```

Value

Returns the license information for WhiteboxTools as an R character vector.
### Description

Creates a block-maximum raster from an input LAS file. When the input/output parameters are not specified, the tool grids all LAS files contained within the working directory.

### Usage

```r
wbt_lidar_block_maximum(
  input,
  output = NULL,
  resolution = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

- **input**: Input LiDAR file.
- **output**: Output file.
- **resolution**: Output raster’s grid resolution.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

### Value

Returns the tool text outputs.
wbt_lidar_block_minimum

Lidar block minimum

Description

Creates a block-minimum raster from an input LAS file. When the input/output parameters are not specified, the tool grids all LAS files contained within the working directory.

Usage

wbt_lidar_block_minimum(
    input,
    output = NULL,
    resolution = 1,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

- input: Input LiDAR file.
- output: Output file.
- resolution: Output raster’s grid resolution.
- wd: Changes the working directory.
- verbose_mode: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- compress_rasters: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- command_only: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
**Description**

Classifies the values in one LiDAR point cloud that correspond with points in a subset cloud.

**Usage**

```r
wbt_lidar_classify_subset(
  base,
  subset,
  output,
  subset_class,
  nonsubset_class = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `base` Input base LiDAR file.
- `subset` Input subset LiDAR file.
- `output` Output LiDAR file.
- `subset_class` Subset point class value (must be 0-18; see LAS specifications).
- `nonsubset_class` Non-subset point class value (must be 0-18; see LAS specifications).
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_lidar_colourize**  
*Lidar colourize*

**Description**

Adds the red-green-blue colour fields of a LiDAR (LAS) file based on an input image.

**Usage**

```r
wbt_lidar_colourize(
  in_lidar,  
  in_image,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

- **in_lidar**: Input LiDAR file.
- **in_image**: Input colour image file.
- **output**: Output LiDAR file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
Description
This tool creates a vector contour coverage from an input LiDAR point file.

Usage

```r
wbt_lidar_contour(
  input,  # Name of the input LiDAR points.
  output = NULL,  # Name of the output vector lines file.
  interval = 10,  # Contour interval.
  smooth = 5,  # Smoothing filter size (in num. points), e.g. 3, 5, 7, 9, 11.
  parameter = "elevation",  # Interpolation parameter; options are ‘elevation’ (default), ‘intensity’, ‘user_data’.
  returns = "all",  # Point return types to include; options are ‘all’ (default), ‘last’, ‘first’.
  exclude_cls = NULL,  # Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, –exclude_cls=’3,4,5,6,7,18’.
  minz = NULL,  # Optional minimum elevation for inclusion in interpolation.
  maxz = NULL,  # Optional maximum elevation for inclusion in interpolation.
  max_triangle_edge_length = NULL,  # Optional maximum triangle edge length; triangles larger than this size will not be gridded.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # If TRUE, will create compressed rasters.
  command_only = FALSE)
```

Arguments

- **input**: Name of the input LiDAR points.
- **output**: Name of the output vector lines file.
- **interval**: Contour interval.
- **smooth**: Smoothing filter size (in num. points), e.g. 3, 5, 7, 9, 11.
- **parameter**: Interpolation parameter; options are ‘elevation’ (default), ‘intensity’, ‘user_data’.
- **returns**: Point return types to include; options are ‘all’ (default), ‘last’, ‘first’.
- **exclude_cls**: Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, –exclude_cls=’3,4,5,6,7,18’.
- **minz**: Optional minimum elevation for inclusion in interpolation.
- **maxz**: Optional maximum elevation for inclusion in interpolation.
- **max_triangle_edge_length**: Optional maximum triangle edge length; triangles larger than this size will not be gridded.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

Usage

```
wb_lidar_digital_surface_model(
    input,
    output = NULL,
    resolution = 1,
    radius = 0.5,
    minz = NULL,
    maxz = NULL,
    max_triangle_edge_length = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

Arguments

- **input**: Input LiDAR file (including extension).
- **output**: Output raster file (including extension).
- **resolution**: Output raster’s grid resolution.
- **radius**: Search Radius.
- **minz**: Optional minimum elevation for inclusion in interpolation.
- **maxz**: Optional maximum elevation for inclusion in interpolation.
- **max_triangle_edge_length**: Optional maximum triangle edge length; triangles larger than this size will not be gridded.
- **wd**: Changes the working directory.

Description

Creates a top-surface digital surface model (DSM) from a LiDAR point cloud.
verbose_mode   Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters       Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only         Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_lidar_elevation_slice**

*Lidar elevation slice*

**Description**

Outputs all of the points within a LiDAR (LAS) point file that lie between a specified elevation range.

**Usage**

```r
wbt_lidar_elevation_slice(
   input,
   output,
   minz = NULL,
   maxz = NULL,
   cls = FALSE,
   inclassval = 2,
   outclassval = 1,
   wd = NULL,
   verbose_mode = FALSE,
   compress_rasters = FALSE,
   command_only = FALSE
)
```

**Arguments**

- **input**: Input LiDAR file.
- **output**: Output LiDAR file.
- **minz**: Minimum elevation value (optional).
- **maxz**: Maximum elevation value (optional).
- **cls**: Optional boolean flag indicating whether points outside the range should be retained in output but reclassified.
- **inclassval**: Optional parameter specifying the class value assigned to points within the slice.
wbt_lidar_ground_point_filter

Lidar ground point filter

Description

Identifies ground points within LiDAR dataset using a slope-based method.

Usage

wbt_lidar_ground_point_filter(
  input, output,
  radius = 2,
  min_neighbours = 0,
  slope_threshold = 45,
  height_threshold = 1,
  classify = TRUE,
  slope_norm = TRUE,
  height_above_ground = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input Input LiDAR file.
output Output LiDAR file.
radius Search Radius.

Value

Returns the tool text outputs.
The minimum number of neighbouring points within search areas. If fewer points than this threshold are identified during the fixed-radius search, a subsequent kNN search is performed to identify the k number of neighbours.

slope_threshold
 Maximum inter-point slope to be considered an off-terrain point.

height_threshold
 Inter-point height difference to be considered an off-terrain point.

classify
 Classify points as ground (2) or off-ground (1).

slope_norm
 Perform initial ground slope normalization?.

height_above_ground
 Transform output to height above average ground elevation?.

wd
 Changes the working directory.

verbose_mode
 Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters
 Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
 Return command that would be executed by system() rather than running tool.

Value
 Returns the tool text outputs.

---

**Description**

Hex-bins a set of LiDAR points.

**Usage**

```r
wbt_lidar_hex_binning(
  input,
  output,
  width,
  orientation = "horizontal",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
Arguments

input  Input LiDAR file.
output Output file.
azimuth Illumination source azimuth in degrees.
altitude Illumination source altitude in degrees.

Value

Returns the tool text outputs.

Description

Calculates a hillshade value for points within a LAS file and stores these data in the RGB field.

Usage

wbt_lidar_hillshade(
  input, output,
  azimuth = 315,
  altitude = 30,
  radius = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input  Input base file.
output Output vector polygon file.
width The grid cell width.
orientation Grid Orientation, ‘horizontal’ or ‘vertical’.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.


**wbt_lidar_histogram**

<table>
<thead>
<tr>
<th>Radius</th>
<th>Search Radius.</th>
</tr>
</thead>
<tbody>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.

---

**wbt_lidar_histogram**  
*Lidar histogram*

**Description**

Creates a histogram of LiDAR data.

**Usage**

```r
wbt_lidar_histogram(
  input,  
  output, 
  parameter = "elevation", 
  clip = 1, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>input</th>
<th>Input LiDAR file.</th>
</tr>
</thead>
<tbody>
<tr>
<td>output</td>
<td>Output HTML file (default name will be based on input file if unspecified).</td>
</tr>
<tr>
<td>parameter</td>
<td>Parameter; options are 'elevation' (default), 'intensity', 'scan angle', 'class', 'time'.</td>
</tr>
<tr>
<td>clip</td>
<td>Amount to clip distribution tails (in percent).</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>
Value

Returns the tool text outputs.

Description

Interpolates LAS files using an inverse-distance weighted (IDW) scheme. When the input/output parameters are not specified, the tool interpolates all LAS files contained within the working directory.

Usage

```r
wbt_lidar_idw_interpolation(
  input,
  output = NULL,
  parameter = "elevation",
  returns = "all",
  resolution = 1,
  weight = 1,
  radius = 2.5,
  exclude_cls = NULL,
  minz = NULL,
  maxz = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input LiDAR file (including extension).
- **output**: Output raster file (including extension).
- **parameter**: Interpolation parameter; options are 'elevation' (default), 'intensity', 'class', 'return_number', 'number_of_returns', 'scan angle', 'rgb', 'user data'.
- **returns**: Point return types to include; options are 'all' (default), 'last', 'first'.
- **resolution**: Output raster's grid resolution.
- **weight**: IDW weight value.
- **radius**: Search Radius.
- **exclude_cls**: Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, --exclude_cls='3,5,6,7,18'.
wbt_lidar_info

minz  Optional minimum elevation for inclusion in interpolation.
maxz  Optional maximum elevation for inclusion in interpolation.
wd    Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

Description

Prints information about a LiDAR (LAS) dataset, including header, point return frequency, and classification data and information about the variable length records (VLRs) and geokeys.

Usage

wbt_lidar_info(
  input,
  output = NULL,
  vlr = TRUE,
  geokeys = TRUE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input  Input LiDAR file.
output  Output HTML file for summary report.
vlr    Flag indicating whether or not to print the variable length records (VLRs).
geokeys  Flag indicating whether or not to print the geokeys.
wd     Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by `system()` rather than running tool.

Value
Returns the tool text outputs.

---

**wbt_lidar_join**  
*Lidar join*

**Description**
Joins multiple LiDAR (LAS) files into a single LAS file.

**Usage**

```r
wbt_lidar_join(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **inputs**: Input LiDAR files.
- **output**: Output LiDAR file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**
Returns the tool text outputs.
**Description**

Performs a kappa index of agreement (KIA) analysis on the classifications of two LAS files.

**Usage**

```R
wbt_lidar_kappa_index(
  input1,  # Input LiDAR classification file.
  input2,  # Input LiDAR reference file.
  output,  # Output HTML file.
  class_accuracy,  # Output classification accuracy raster file.
  resolution = 1,  # Output raster's grid resolution.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE  # Return command that would be executed by system() rather than running tool.
)
```

**Arguments**

- `input1`: Input LiDAR classification file.
- `output`: Output HTML file.
- `class_accuracy`: Output classification accuracy raster file.
- `resolution`: Output raster’s grid resolution.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_lidar_nearest_neighbour_gridding

Lidar nearest neighbour gridding

Description

Grids LiDAR files using nearest-neighbour scheme. When the input/output parameters are not specified, the tool grids all LAS files contained within the working directory.

Usage

```r
wbt_lidar_nearest_neighbour_gridding(
  input,
  output = NULL,
  parameter = "elevation",
  returns = "all",
  resolution = 1,
  radius = 2.5,
  exclude_cls = NULL,
  minz = NULL,
  maxz = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input LiDAR file (including extension).
- **output**: Output raster file (including extension).
- **parameter**: Interpolation parameter; options are 'elevation' (default), 'intensity', 'class', 'return_number', 'number_of_returns', 'scan angle', 'rgb', 'user data'.
- **returns**: Point return types to include; options are 'all' (default), 'last', 'first'.
- **resolution**: Output raster's grid resolution.
- **radius**: Search Radius.
- **exclude_cls**: Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, –exclude_cls=’3,4,5,6,7,18’.
- **minz**: Optional minimum elevation for inclusion in interpolation.
- **maxz**: Optional maximum elevation for inclusion in interpolation.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.


**wbt_lidar_point_density**

*Lidar point density*

**Description**

Calculates the spatial pattern of point density for a LiDAR data set. When the input/output parameters are not specified, the tool grids all LAS files contained within the working directory.

**Usage**

```r
wbt_lidar_point_density(
  input,
  output = NULL,
  returns = "all",
  resolution = 1,
  radius = 2.5,
  exclude_cls = NULL,
  minz = NULL,
  maxz = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input LiDAR file (including extension).
- **output**: Output raster file (including extension).
- **returns**: Point return types to include; options are 'all' (default), 'last', 'first'.
- **resolution**: Output raster's grid resolution.
- **radius**: Search radius.
- **exclude_cls**: Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, -exclude_cls='3,4,5,6,7,18'.
- **minz**: Optional minimum elevation for inclusion in interpolation.

**compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

**command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_lidar_point_return_analysis

Lidar point return analysis

Description

This tool performs a quality control check on the return values of points in a LiDAR file.

Usage

```r
wbt_lidar_point_return_analysis(
  input,
  output = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Name of the input LiDAR points.
- **output**: Name of the output LiDAR points.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**Lidar point stats**

**Description**

Creates several rasters summarizing the distribution of LAS point data. When the input/output parameters are not specified, the tool works on all LAS files contained within the working directory.

**Usage**

```r
wbt_lidar_point_stats(
  input,
  resolution = 1,
  num_points = TRUE,
  num_pulses = FALSE,
  avg_points_per_pulse = TRUE,
  z_range = FALSE,
  intensity_range = FALSE,
  predom_class = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input`: Input LiDAR file.
- `resolution`: Output raster’s grid resolution.
- `num_points`: Flag indicating whether or not to output the number of points (returns) raster.
- `num_pulses`: Flag indicating whether or not to output the number of pulses raster.
- `avg_points_per_pulse`: Flag indicating whether or not to output the average number of points (returns) per pulse raster.
- `z_range`: Flag indicating whether or not to output the elevation range raster.
- `intensity_range`: Flag indicating whether or not to output the intensity range raster.
- `predom_class`: Flag indicating whether or not to output the predominant classification raster.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.
Value

Returns the tool text outputs.

wbt_lidar_ransac_planes

Lidar ransac planes

Description

Performs a RANSAC analysis to identify points within a LiDAR point cloud that belong to linear planes.

Usage

```r
wbt_lidar_ransac_planes(
  input, output,
  radius = 2, num_iter = 50, num_samples = 5,
  threshold = 0.35, model_size = 8, max_slope = 80,
  classify = FALSE, last_returns = FALSE, wd = NULL,
  verbose_mode = FALSE, compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input LiDAR file.
- **output**: Output LiDAR file.
- **radius**: Search Radius.
- **num_iter**: Number of iterations.
- **num_samples**: Number of sample points on which to build the model.
- **threshold**: Threshold used to determine inlier points.
- **model_size**: Acceptable model size.
- **max_slope**: Maximum planar slope.
- **classify**: Classify points as ground (2) or off-ground (1).
- **last_returns**: Only include last- and only-return points.
wd Changes the working directory.

verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Interpolates LAS files using a radial basis function (RBF) scheme. When the input/output parameters are not specified, the tool interpolates all LAS files contained within the working directory.

Usage

wbt_lidar_rbf_interpolation(
  input,
  output = NULL,
  parameter = "elevation",
  returns = "all",
  resolution = 1,
  num_points = 20,
  exclude_cls = NULL,
  minz = NULL,
  maxz = NULL,
  func_type = "ThinPlateSpline",
  poly_order = "none",
  weight = 5,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
**Arguments**

- **input**: Input LiDAR file (including extension).
- **output**: Output raster file (including extension).
- **parameter**: Interpolation parameter; options are 'elevation' (default), 'intensity', 'class', 'return_number', 'number_of_returns', 'scan angle', 'rgb', 'user data'.
- **returns**: Point return types to include; options are 'all' (default), 'last', 'first'.
- **resolution**: Output raster's grid resolution.
- **num_points**: Number of points.
- **exclude_cls**: Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, –exclude_cls='3,4,5,6,7,18'.
- **minz**: Optional minimum elevation for inclusion in interpolation.
- **maxz**: Optional maximum elevation for inclusion in interpolation.
- **func_type**: Radial basis function type; options are 'ThinPlateSpline' (default), 'PolyHarmonic', 'Gaussian', 'MultiQuadric', 'InverseMultiQuadric'.
- **poly_order**: Polynomial order; options are 'none' (default), 'constant', 'affine'.
- **weight**: Weight parameter used in basis function.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_lidar_remove_duplicates**

*Lidar remove duplicates*

**Description**

Removes duplicate points from a LiDAR data set.
Usage

```
wbt_lidar_remove_duplicates(
    input,
    output,
    include_z = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

Arguments

- **input**: Input LiDAR file.
- **output**: Output LiDAR file.
- **include_z**: Include z-values in point comparison?.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_lidar_remove_outliers**

*Lidar remove outliers*

Description

Removes outliers (high and low points) in a LiDAR point cloud.

Usage

```
wbt_lidar_remove_outliers(
    input,
    output,
    radius = 2,
    elev_diff = 50,
    use_median = FALSE,
    classify = TRUE,
)```
wbt_lidar_rooftop_analysis

Lidar rooftop analysis

Description

Identifies roof segments in a LiDAR point cloud.

Usage

wbt_lidar_rooftop_analysis(
    buildings,
    output,
    input = NULL,
    radius = 2,
    num_iter = 50,
    num_samples = 10,
    threshold = 0.15,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```r
model_size = 15,
max_slope = 65,
norm_diff = 10,
azimuth = 180,
altitude = 30,
wd = NULL,
verbose_mode = FALSE,
compress_rasters = FALSE,
command_only = FALSE
)
```

**Arguments**

- **buildings**
  - Input vector build footprint polygons file.

- **output**
  - Output vector polygon file.

- **input**
  - Input LiDAR file.

- **radius**
  - Search Radius.

- **num_iter**
  - Number of iterations.

- **num_samples**
  - Number of sample points on which to build the model.

- **threshold**
  - Threshold used to determine inlier points (in elevation units).

- **model_size**
  - Acceptable model size, in points.

- **max_slope**
  - Maximum planar slope, in degrees.

- **norm_diff**
  - Maximum difference in normal vectors, in degrees.

- **azimuth**
  - Illumination source azimuth, in degrees.

- **altitude**
  - Illumination source altitude in degrees.

- **wd**
  - Changes the working directory.

- **verbose_mode**
  - Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

- **compress_rasters**
  - Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

- **command_only**
  - Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.
wbt_lidar_segmentation

Lidar segmentation

Description

Segments a LiDAR point cloud based on differences in the orientation of fitted planar surfaces and point proximity.

Usage

wbt_lidar_segmentation(
    input,
    output,
    radius = 2,
    num_iter = 50,
    num_samples = 10,
    threshold = 0.15,
    model_size = 15,
    max_slope = 80,
    norm_diff = 10,
    maxzdiff = 1,
    classes = FALSE,
    ground = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input        Input LiDAR file.
output       Output LiDAR file.
radius       Search Radius.
num_iter     Number of iterations.
num_samples  Number of sample points on which to build the model.
threshold    Threshold used to determine inlier points.
model_size   Acceptable model size.
max_slope    Maximum planar slope.
norm_diff    Maximum difference in normal vectors, in degrees.
maxzdiff     Maximum difference in elevation (z units) between neighbouring points of the same segment.
classes      Segments don’t cross class boundaries.
Classify the largest segment as ground points?.

Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

Return command that would be executed by system() rather than running tool.

Returns the tool text outputs.

Identifies ground points within LiDAR point clouds using a segmentation based approach.

Input LiDAR file.

Output file.

Search Radius.

Maximum difference in normal vectors, in degrees.

Maximum difference in elevation (z units) between neighbouring points of the same segment.

Classify points as ground (2) or off-ground (1).

Description

Usage

Arguments
wbt_lidar_shift

wd
verbose_mode
compress_rasters
command_only

Changes the working directory.
Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_lidar_shift  Lidar shift

Description

Shifts the x,y,z coordinates of a LiDAR file.

Usage

```r
wbt_lidar_shift(
  input,
  output,
  x_shift = "",
  y_shift = "",
  z_shift = "",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Name of the input LiDAR points.
- **output**: Name of the output LiDAR points.
- **x_shift**: x-shift value, blank for none.
- **y_shift**: y-shift value, blank for none.
- **z_shift**: z-shift value, blank for none.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.
**Value**

Returns the tool text outputs.

---

**Description**

This tool interpolates one or more LiDAR tiles using Sibson’s natural neighbour method.

**Usage**

```r
wbt_lidar_sibson_interpolation(
  input,
  output = NULL,
  parameter = "elevation",
  returns = "all",
  resolution = 1,
  exclude_cls = NULL,
  minz = NULL,
  maxz = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Name of the input LiDAR points (leave blank to use all files in WorkingDirectory).
- **output**: Output raster file (including extension).
- **parameter**: Interpolation parameter; options are 'elevation' (default), 'intensity', 'class', 'return_number', 'number_of_returns', 'scan angle', 'user_data'.
- **returns**: Point return types to include; options are 'all' (default), 'last', 'first'.
- **resolution**: Output raster’s grid resolution.
- **exclude_cls**: Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, `exclude_cls='3,4,5,6,7,18'`.
- **minz**: Optional minimum elevation for inclusion in interpolation.
- **maxz**: Optional maximum elevation for inclusion in interpolation.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Whether to compress rasters.
wbt_lidar_sort_by_time

Lidar sort by time

Description

This sorts the points in a LiDAR file by the GPS time.

Usage

wbt_lidar_sort_by_time(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input Name of the input LiDAR points.
output Name of the output LiDAR points.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
**Description**

Thins a LiDAR point cloud, reducing point density.

**Usage**

```r
wbt_lidar_thin(
  input,
  output,
  resolution = 2,
  method = "lowest",
  save_filtered = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input`: Input LiDAR file.
- `output`: Output LiDAR file.
- `resolution`: The size of the square area used to evaluate nearby points in the LiDAR data.
- `method`: Point selection method; options are 'first', 'last', 'lowest' (default), 'highest', 'nearest'.
- `save_filtered`: Save filtered points to separate file?.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_lidar_thin_high_density**

*Lidar thin high density*

**Description**

Thins points from high density areas within a LiDAR point cloud.

**Usage**

```r
wbt_lidar_thin_high_density(
  input,
  output,
  density,
  resolution = 1,
  save_filtered = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**  
  Input LiDAR file.
- **output**  
  Output LiDAR file.
- **density**  
  Max. point density (points / m^3).
- **resolution**  
  Output raster’s grid resolution.
- **save_filtered**  
  Save filtered points to separate file?.
- **wd**  
  Changes the working directory.
- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**  
  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_lidar_tile**

**Lidar tile**

**Description**

Tiles a LiDAR LAS file into multiple LAS files.

**Usage**

```r
wbt_lidar_tile(
  input,  
  width = 1000, 
  height = 1000, 
  origin_x = 0, 
  origin_y = 0, 
  min_points = 2, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)
```

**Arguments**

- **input**  
  Input LiDAR file.
- **width**  
  Width of tiles in the X dimension; default 1000.0.
- **height**  
  Height of tiles in the Y dimension.
- **origin_x**  
  Origin point X coordinate for tile grid.
- **origin_y**  
  Origin point Y coordinate for tile grid.
- **min_points**  
  Minimum number of points contained in a tile for it to be saved.
- **wd**  
  Changes the working directory.
- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**  
  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_lidar_tile_footprint**

*Lidar tile footprint*

**Description**

Creates a vector polygon of the convex hull of a LiDAR point cloud. When the input/output parameters are not specified, the tool works with all LAS files contained within the working directory.

**Usage**

```r
wbt_lidar_tile_footprint(
  input,
  output,
  hull = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input LiDAR file.
- **output**: Output vector polygon file.
- **hull**: Identify the convex hull around points.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**Lidar tin gridding**

**Description**

Creates a raster grid based on a Delaunay triangular irregular network (TIN) fitted to LiDAR points.

**Usage**

```r
wbt_lidar_tin_gridding(
  input, 
  output = NULL, 
  parameter = "elevation", 
  returns = "all", 
  resolution = 1, 
  exclude_cls = "7,18", 
  minz = NULL, 
  maxz = NULL, 
  max_triangle_edge_length = NULL, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)
```

**Arguments**

- **input**: Input LiDAR file (including extension).
- **output**: Output raster file (including extension).
- **parameter**: Interpolation parameter; options are 'elevation' (default), 'intensity', 'class', 'return_number', 'number_of_returns', 'scan angle', 'rgb', 'user data'.
- **returns**: Point return types to include; options are 'all' (default), 'last', 'first'.
- **resolution**: Output raster's grid resolution.
- **exclude_cls**: Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, -exclude_cls=’3,4,5,6,7,18’.
- **minz**: Optional minimum elevation for inclusion in interpolation.
- **maxz**: Optional maximum elevation for inclusion in interpolation.
- **max_triangle_edge_length**: Optional maximum triangle edge length; triangles larger than this size will not be gridded.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only
Return command that would be executed by `system()` rather than running tool.

Value
Returns the tool text outputs.

---

**wbt_lidar_tophat_transform**

*Lidar tophat transform*

### Description
Performs a white top-hat transform on a Lidar dataset; as an estimate of height above ground, this is useful for modelling the vegetation canopy.

### Usage
```r
wbt_lidar_tophat_transform(
  input,
  output,
  radius = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments
- **input** Input LiDAR file.
- **output** Output LiDAR file.
- **radius** Search Radius.
- **wd** Changes the working directory.
- **verbose_mode** Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters** Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only** Return command that would be executed by `system()` rather than running tool.

### Value
Returns the tool text outputs.
wbt_linearity_index  Linearity index

Description

Calculates the linearity index for vector polygons.

Usage

wbt_linearity_index(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input  Input vector polygon file.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_lines_to_polygons  Lines to polygons

Description

Converts vector polylines to polygons.
Usage

\texttt{wbt\_lines\_to\_polygons(}
\begin{verbatim}
   input,
   output,
   wd = NULL,
   verbose\_mode = FALSE,
   compress\_rasters = FALSE,
   command\_only = FALSE
\end{verbatim}
\texttt{)}

Arguments

\begin{itemize}
\item \texttt{input} Input vector line file.
\item \texttt{output} Output vector polygon file.
\item \texttt{wd} Changes the working directory.
\item \texttt{verbose\_mode} Sets verbose mode. If verbose mode is \texttt{FALSE}, tools will not print output messages.
\item \texttt{compress\_rasters} Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
\item \texttt{command\_only} Return command that would be executed by \texttt{system()} rather than running tool.
\end{itemize}

Value

Returns the tool text outputs.

wbt\_line\_detection\_filter

\textit{Line detection filter}

Description

Performs a line-detection filter on an image.

Usage

\texttt{wbt\_line\_detection\_filter(}
\begin{verbatim}
   input,
   output,
   variant = "vertical",
   absvals = FALSE,
   clip = 0,
   wd = NULL,
   verbose\_mode = FALSE,
   compress\_rasters = FALSE,
   command\_only = FALSE
\end{verbatim}
\texttt{)}
**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **variant**: Optional variant value. Options include 'v' (vertical), 'h' (horizontal), '45', and '135' (default is 'v').
- **absvals**: Optional flag indicating whether outputs should be absolute values.
- **clip**: Optional amount to clip the distribution tails by, in percent.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**Description**

Identifies points where the features of two vector line layers intersect.

**Usage**

```r
wbt_line_intersections(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
**Arguments**

- **input1**  
  Input vector polyline file.
- **input2**  
  Input vector polyline file.
- **output**  
  Output vector point file.
- **wd**  
  Changes the working directory.
- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**  
  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_line_thinning**

**Line thinning**

**Description**

Performs line thinning a on Boolean raster image; intended to be used with the RemoveSpurs tool.

**Usage**

```r
wbt_line_thinning(
  input,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)
```

**Arguments**

- **input**  
  Input raster file.
- **output**  
  Output raster file.
- **wd**  
  Changes the working directory.
- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**  
  Return command that would be executed by `system()` rather than running tool.
Value

Returns the tool text outputs.

Description

All available tools in WhiteboxTools

Usage

wbt_list_tools(keywords = NULL)

Arguments

keywords Keywords may be used to search available tools.

Value

Return all available tools in WhiteboxTools that contain the keywords.

Examples

## Not run:
wbt_list_tools("lidar")

## End(Not run)

Description

List unique values

Lists the unique values contained in a field within a vector’s attribute table.

Usage

wbt_list_unique_values(
    input,
    field,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input  Input raster file.
field  Input field name in attribute table.
output  Output HTML file (default name will be based on input file if unspecified).
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Returns the natural logarithm of values in a raster.

Usage

```r
wbt_ln(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

input  Input raster file.
output  Output raster file.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.
**Value**

Returns the tool text outputs.

**Description**

This tool calculates a local, neighbourhood-based hypsometric integral raster.

**Usage**

```r
wbt_local_hypsometric_analysis(
    input,
    out_mag,
    out_scale,
    min_scale = 4,
    step = 1,
    num_steps = 10,
    step_nonlinearity = 1,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

- **input** Name of the input raster DEM file.
- **out_mag** Name of the openness output raster file.
- **out_scale** Name of the openness output raster file.
- **min_scale** Minimum search neighbourhood radius in grid cells.
- **step** Step size as any positive non-zero integer.
- **num_steps** Number of steps.
- **step_nonlinearity** Step nonlinearity factor (1.0-2.0 is typical).
- **wd** Changes the working directory.
- **verbose_mode** Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters** Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only** Return command that would be executed by `system()` rather than running tool.
wbt_local_quadratic_regression

Local quadratic regression

Description

This tool is an implementation of the constrained quadratic regression algorithm using a flexible window size described in Wood (1996).

Usage

wbt_local_quadratic_regression(
    dem,
    output,
    filter = 3,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

- **dem**: Name of the input DEM raster file.
- **output**: Name of the output raster file.
- **filter**: Edge length of the filter kernel.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
**wbt_log10**

*Log10*

---

**Description**

Returns the base-10 logarithm of values in a raster.

**Usage**

```r
wbt_log10(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_log2**

*Log2*

---

**Description**

Returns the base-2 logarithm of values in a raster.
Usage

```r
wbt_log2(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

### wbt_logistic_regression

Logistic regression

Description

Performs a logistic regression analysis using training site polygons/points and predictor rasters.

Usage

```r
wbt_logistic_regression(
  inputs,
  training,
  field,
  scaling = "Normalize",
  output = NULL,
  test_proportion = 0.2,
  wd = NULL,
  verbose_mode = FALSE,
)
wbt_longest_flowpath

    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

inputs          Names of the input predictor rasters.
training        Name of the input training site polygons/points shapefile.
field           Name of the attribute containing class data.
scaling         Scaling method for predictors. Options include 'None', 'Normalize', and 'Standardize'.
output          Name of the output raster file.
test_proportion The proportion of the dataset to include in the test split; default is 0.2.
wd              Changes the working directory.
verbose_mode    Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only    Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

wbt_longest_flowpath  Longest flowpath

Description

Delineates the longest flowpaths for a group of subbasins or watersheds.

Usage

wbt_longest_flowpath(
    dem,
    basins,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)


Arguments

dem  Input raster DEM file.
basins Input raster basins file.
output Output vector file.
wd   Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_long_profile  Long profile

Description

Plots the stream longitudinal profiles for one or more rivers.

Usage

wbt_long_profile(
  d8_pntr,
  streams,
  dem,
  output,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

d8_pntr  Input raster D8 pointer file.
streams  Input raster streams file.
dem  Input raster DEM file.
output  Output HTML file.
esri_pntr  D8 pointer uses the ESRI style scheme.
**Description**

Plots the longitudinal profiles from flow-paths initiating from a set of vector points.

**Usage**

```r
wbt_long_profile_from_points(
  d8_pntr,
  points,
  dem,
  output,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `d8_pntr` Input raster D8 pointer file.
- `points` Input vector points file.
- `dem` Input raster DEM file.
- `output` Output HTML file.
- `esri_pntr` D8 pointer uses the ESRI style scheme.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.
wbt_lowest_position

compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

wbt_lowest_position  Lowest position

Description
Identifies the stack position of the minimum value within a raster stack on a cell-by-cell basis.

Usage
wbt_lowest_position(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments
inputs  Input raster files.
output  Output raster file.
wd      Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only    Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.
Description

This tool locates saddle points along ridges within a digital elevation model (DEM).

Usage

```r
wbt_low_points_on_headwater_divides(
  dem,
  streams,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

dem Name of the input DEM raster file.
streams Name of the input stream channel raster file.
output Name of the output vector file.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_majority_filter  Majority filter

Description

Assigns each cell in the output grid the most frequently occurring value (mode) in a moving window centred on each grid cell in the input raster.

Usage

wbt_majority_filter(
  input,
  output,
  filterx = 11,
  filtery = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input        Input raster file.
output       Output raster file.
filterx      Size of the filter kernel in the x-direction.
filtery      Size of the filter kernel in the y-direction.
wd           Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
Map off terrain objects

Maps off-terrain objects in a digital elevation model (DEM).

Usage

```r
wbt_map_off_terrain_objects(
  dem,          
  output,       
  max_slope = 40,  
  min_size = 1,   
  wd = NULL,     
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)
```

Arguments

output: Output raster file.
max_slope: Maximum inter-cell absolute slope.
min_size: Minimum feature size, in grid cells.
wd: Changes the working directory.
verbose_mode: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**wbt_max**

**Max**

**Description**

Performs a MAX operation on two rasters or a raster and a constant value.

**Usage**

```r
wbt_max(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input1`  Input raster file or constant value.
- `input2`  Input raster file or constant value.
- `output`  Output raster file.
- `wd`  Changes the working directory.
- `verbose_mode`  Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_maximal_curvature**  
*Maximal curvature*

**Description**

Calculates a mean curvature raster from an input DEM.

**Usage**

```r
wbt_maximal_curvature(
  dem,  
  output,  
  log = FALSE,  
  zfactor = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)
```

**Arguments**

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **log**: Display output values using a log-scale.
- **zfactor**: Optional multiplier for when the vertical and horizontal units are not the same.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_maximum_filter

**Maximum filter**

**Description**

Assigns each cell in the output grid the maximum value in a moving window centred on each grid cell in the input raster.

**Usage**

```r
wbt_maximum_filter(
  input,  # Input raster file.
  output,  # Output raster file.
  filterx = 11,  # Size of the filter kernel in the x-direction.
  filtery = 11,  # Size of the filter kernel in the y-direction.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE  # Return command that would be executed by system() rather than running tool.
)
```

**Arguments**

- `output`: Output raster file.
- `filterx`: Size of the filter kernel in the x-direction.
- `filtery`: Size of the filter kernel in the y-direction.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
\textbf{wbt_max_absolute_overlay}

\textit{Max absolute overlay}

\textbf{Description}

Evaluates the maximum absolute value for each grid cell from a stack of input rasters.

\textbf{Usage}

\begin{verbatim}
wbt_max_absolute_overlay(
    inputs,  
    output,  
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
\end{verbatim}

\textbf{Arguments}

- \textbf{inputs} \hspace{1cm} Input raster files.
- \textbf{output} \hspace{1cm} Output raster file.
- \textbf{wd} \hspace{1cm} Changes the working directory.
- \textbf{verbose_mode} \hspace{1cm} Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- \textbf{compress_rasters} \hspace{1cm} Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- \textbf{command_only} \hspace{1cm} Return command that would be executed by \texttt{system()} rather than running tool.

\textbf{Value}

Returns the tool text outputs.

\textbf{wbt_max_anisotropy_dev}

\textit{Max anisotropy dev}

\textbf{Description}

Calculates the maximum anisotropy (directionality) in elevation deviation over a range of spatial scales.
Usage

```r
wbt_max_anisotropy_dev(
  dem,
  out_mag,
  out_scale,
  max_scale,
  min_scale = 3,
  step = 2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **dem**: Input raster DEM file.
- **out_mag**: Output raster DEVmax magnitude file.
- **out_scale**: Output raster DEVmax scale file.
- **max_scale**: Maximum search neighbourhood radius in grid cells.
- **min_scale**: Minimum search neighbourhood radius in grid cells.
- **step**: Step size as any positive non-zero integer.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_max_anisotropy_dev_signature**

*Max anisotropy dev signature*

---

Description

Calculates the anisotropy in deviation from mean for points over a range of spatial scales.
**Usage**

```r
wbt_max_anisotropy_dev_signature(
  dem,
  points,
  output,
  max_scale,
  min_scale = 1,
  step = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `points` : Input vector points file.
- `output` : Output HTML file.
- `max_scale` : Maximum search neighbourhood radius in grid cells.
- `min_scale` : Minimum search neighbourhood radius in grid cells.
- `step` : Step size as any positive non-zero integer.
- `wd` : Changes the working directory.
- `verbose_mode` : Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` : Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` : Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

**Description**

Lindsay and Seibert’s (2013) branch length index is used to map drainage divides or ridge lines.
Usage

```r
wbt_max_branch_length(
  dem,
  output,
  log = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `output`: Output raster file.
- `log`: Optional flag to request the output be log-transformed.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If `verbose_mode` is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

**Description**

Calculates the maximum difference from mean elevation over a range of spatial scales.

Usage

```r
wbt_max_difference_from_mean(
  dem,
  out_mag,
  out_scale,
  min_scale,
  max_scale,
  step = 1,
)```
wbt_max_downslope_elev_change

wd = NULL,
verbose_mode = FALSE,
compress_rasters = FALSE,
command_only = FALSE
)

Arguments

dem Input raster DEM file.
out_mag Output raster DIFFmax magnitude file.
out_scale Output raster DIFFmax scale file.
min_scale Minimum search neighbourhood radius in grid cells.
max_scale Maximum search neighbourhood radius in grid cells.
step Step size as any positive non-zero integer.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_max_downslope_elev_change

Max downslope elev change

Description

Calculates the maximum downslope change in elevation between a grid cell and its eight downslope neighbors.

Usage

wbt_max_downslope_elev_change(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
Arguments

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_max_elevation_deviation**

*Max elevation deviation*

Description

Calculates the maximum elevation deviation over a range of spatial scales.

Usage

```r
wbt_max_elevation_deviation(
  dem,  
  out_mag,  
  out_scale,  
  min_scale,  
  max_scale,  
  step = 1,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)
```

Arguments

- **dem**: Input raster DEM file.
- **out_mag**: Output raster DEVmax magnitude file.
- **out_scale**: Output raster DEVmax scale file.
- **min_scale**: Minimum search neighbourhood radius in grid cells.
wbt_max_elev_dev_signature

Description

Calculates the maximum elevation deviation over a range of spatial scales and for a set of points.

Usage

wbt_max_elev_dev_signature(
  dem,  # Input raster DEM file.
  points,  # Input vector points file.
  output,  # Output HTML file.
  min_scale,  # Minimum search neighbourhood radius in grid cells.
  max_scale,  # Maximum search neighbourhood radius in grid cells.
  step = 10,  # Step size as any positive non-zero integer.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE  # Return command that would be executed by system() rather than running tool.
)

Arguments

- **dem**: Input raster DEM file.
- **points**: Input vector points file.
- **output**: Output HTML file.
- **min_scale**: Minimum search neighbourhood radius in grid cells.
- **max_scale**: Maximum search neighbourhood radius in grid cells.
Description
Evaluates the maximum value for each grid cell from a stack of input rasters.

Usage

```r
wbt_max_overlay(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `inputs`: Input raster files.
- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

- Returns the tool text outputs.
**wbt_max_upslope_elev_change**

*Max upslope elev change*

**Description**

Calculates the maximum upslope change in elevation between a grid cell and its eight downslope neighbors.

**Usage**

```r
calculate_slope_change(
  dem,  # Input raster DEM file.
  output,  # Output raster file.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE  # Return command that would be executed by system() rather than running tool.
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dem</td>
<td>Input raster DEM file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.

---

**wbt_max_upslope_flowpath_length**

*Max upslope flowpath length*

**Description**

Measures the maximum length of all upslope flowpaths draining each grid cell.
Usage

```r
wbt_max_upslope_flowpath_length(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

Arguments

dem Input raster DEM file.
output Output raster file.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_md_inf_flow_accumulation**

*Md inf flow accumulation*

Description

Calculates an FD8 flow accumulation raster from an input DEM.

Usage

```r
wbt_md_inf_flow_accumulation(
    dem,
    output,
    out_type = "specific contributing area",
    exponent = 1.1,
    threshold = NULL,
    log = FALSE,
    clip = FALSE,
    wd = NULL,
```
Arguments

dem Input raster DEM file.
output Output raster file.
out_type Output type; one of 'cells', 'specific contributing area' (default), and 'catchment area'.
extponent Optional exponent parameter; default is 1.1.
threshold Optional convergence threshold parameter, in grid cells; default is infinity.
log Optional flag to request the output be log-transformed.
clip Optional flag to request clipping the display max by 1 percent.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Calculates a mean curvature raster from an input DEM.

Usage

wbt_mean_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
Arguments

- **dem**
  Input raster DEM file.

- **output**
  Output raster file.

- **log**
  Display output values using a log-scale.

- **zfactor**
  Optional multiplier for when the vertical and horizontal units are not the same.

- **wd**
  Changes the working directory.

- **verbose_mode**
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

- **compress_rasters**
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

- **command_only**
  Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_mean_filter**  
**Mean filter**

Description

Performs a mean filter (low-pass filter) on an input image.

Usage

```r
wbt_mean_filter(
  input,
  output,
  filterx = 3,
  filtery = 3,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**
  Input raster file.

- **output**
  Output raster file.

- **filterx**
  Size of the filter kernel in the x-direction.

- **filtery**
  Size of the filter kernel in the y-direction.

- **wd**
  Changes the working directory.
**wbt_median_filter**

- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_median_filter**  *Median filter*

**Description**

Performs a median filter on an input image.

**Usage**

```r
wbt_median_filter(
  input,
  output,
  filterx = 11,
  filtery = 11,
  sig_digits = 2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **filterx**: Size of the filter kernel in the x-direction.
- **filtery**: Size of the filter kernel in the y-direction.
- **sig_digits**: Number of significant digits.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.
`wbt_medoid`  

**Value**

Returns the tool text outputs.

---

**Description**

Calculates the medoid for a series of vector features contained in a shapefile.

**Usage**

```r
wbt_medoid(
  input,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)
```

**Arguments**

- `input` Input vector file.
- `output` Output vector file.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_merge_line_segments**

*Merge line segments*

**Description**

Merges vector line segments into larger features.

**Usage**

```r
wbt_merge_line_segments(
    input, 
    output, 
    snap = 0, 
    wd = NULL, 
    verbose_mode = FALSE, 
    compress_rasters = FALSE, 
    command_only = FALSE 
)
```

**Arguments**

- **input**: Input vector file.
- **output**: Output vector file.
- **snap**: Snap tolerance.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_merge_table_with_csv

*Merge table with csv*

**Description**

Merge a vector’s attribute table with a table contained within a CSV text file.

**Usage**

```r
wbt_merge_table_with_csv(
  input,  
pkey,      
csv,      
fkey,      
import_field = NULL,  
wd = NULL,  
verbose_mode = FALSE, 
compress_rasters = FALSE,  
command_only = FALSE
)
```

**Arguments**

- **input**: Input primary vector file (i.e. the table to be modified).
- **pkey**: Primary key field.
- **csv**: Input CSV file (i.e. source of data to be imported).
- **fkey**: Foreign key field.
- **import_field**: Imported field (all fields will be imported if not specified).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_merge_vectors**  
**Merge vectors**

**Description**
Combines two or more input vectors of the same ShapeType creating a single, new output vector.

**Usage**

```r
wbt_merge_vectors(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**
- **inputs**: Input vector files.
- **output**: Output vector file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**
Returns the tool text outputs.

---

**wbt_min**  
**Min**

**Description**
Performs a MIN operation on two rasters or a raster and a constant value.
Usage

```r
wbt_min(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input1**: Input raster file or constant value.
- **input2**: Input raster file or constant value.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_minimal_curvature**  *Minimal curvature*

Description

Calculates a mean curvature raster from an input DEM.

Usage

```r
wbt_minimal_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dem</td>
<td>Input raster DEM file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>log</td>
<td>Display output values using a log-scale.</td>
</tr>
<tr>
<td>zfactor</td>
<td>Optional multiplier for when the vertical and horizontal units are not the same.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode.  If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>

Value

Returns the tool text outputs.

Description

Creates a vector minimum bounding rectangle around vector features.

Usage

```r
wbt_minimum_bounding_box(
  input,
  output,
  criterion = "area",
  features = TRUE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input vector file.</td>
</tr>
<tr>
<td>output</td>
<td>Output vector polygon file.</td>
</tr>
<tr>
<td>criterion</td>
<td>Minimization criterion; options include 'area' (default), 'length', 'width', and 'perimeter'.</td>
</tr>
</tbody>
</table>
wbt_minimum_bounding_circle

Minimum bounding circle

Description
Delineates the minimum bounding circle (i.e. smallest enclosing circle) for a group of vectors.

Usage
wbt_minimum_bounding_circle(
  input, output,
  features = TRUE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments
input Input vector file.
output Output vector polygon file.
features Find the minimum bounding circle around each individual vector feature.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.
Value

Returns the tool text outputs.

Arguments

- **input**: Input vector file.
- **output**: Output vector polygon file.
- **features**: Find the minimum bounding envelop around each individual vector feature.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_minimum_convex_hull

Minimum convex hull

Description

Creates a vector convex polygon around vector features.

Usage

wbt_minimum_convex_hull(
    input,
    output,
    features = TRUE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input       | Input vector file.
output      | Output vector polygon file.
features    | Find the hulls around each vector feature.
wd          | Changes the working directory.
verbose_mode| Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters | Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only | Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
wbt_minimum_filter

Description

Assigns each cell in the output grid the minimum value in a moving window centred on each grid cell in the input raster.

Usage

wbt_minimum_filter(
    input,
    output,
    filterx = 11,
    filtery = 11,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input        Input raster file.
output       Output raster file.
filterx      Size of the filter kernel in the x-direction.
filtery      Size of the filter kernel in the y-direction.
wd           Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
**wbt_min_absolute_overlay**

*Min absolute overlay*

**Description**
Evaluates the minimum absolute value for each grid cell from a stack of input rasters.

**Usage**
```
wbt_min_absolute_overlay(
    inputs,  # Input raster files.
    output,  # Output raster file.
    wd = NULL,  # Changes the working directory.
    verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
    compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
    command_only = FALSE
)
```

**Arguments**

- **inputs**: Input raster files.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**
Returns the tool text outputs.

---

**wbt_min_dist_classification**

*Min dist classification*

**Description**
Performs a supervised minimum-distance classification using training site polygons and multispectral images.
Usage

```
  wbt_min_dist_classification(
    inputs,
    polys,
    field,
    output,
    threshold = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
  )
```

Arguments

- **inputs**: Names of the input band images.
- **polys**: Name of the input training site polygons shapefile.
- **field**: Name of the attribute containing class name data.
- **output**: Name of the output raster file.
- **threshold**: Distance threshold, in z-scores; blank for none.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Calculates the minimum downslope change in elevation between a grid cell and its eight downslope neighbors.
Usage

wbt_min_donslope_elev_change(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

dem              Input raster DEM file.
output            Output raster file.
wd                Changes the working directory.
verbose_mode      Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only      Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

wbt_min_max_contrast_stretch

Min max contrast stretch

Description

Performs a min-max contrast stretch on an input greytone image.

Usage

wbt_min_max_contrast_stretch(
    input,
    output,
    min_val,
    max_val,
    num_tones = 256,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **min_val**: Lower tail clip value.
- **max_val**: Upper tail clip value.
- **num_tones**: Number of tones in the output image.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

### Description

Evaluates the minimum value for each grid cell from a stack of input rasters.

### Usage

```r
wbt_min_overlay(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

- **inputs**: Input raster files.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
`compress_rasters`  
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

`command_only`  
Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

### `wbt_modified_k_means_clustering`

**Modified k means clustering**

**Description**

Performs a modified k-means clustering operation on a multi-spectral dataset.

**Usage**

```r
wbt_modified_k_means_clustering(
  inputs,  # Input raster files.
  output,  # Output raster file.
  out_html = NULL,  # Output HTML report file.
  start_clusters = 1000,  # Initial number of clusters.
  merge_dist = NULL,  # Cluster merger distance.
  max_iterations = 10,  # Maximum number of iterations.
  class_change = 2,  # Minimum percent of cells changed between iterations before completion.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Return command that would be executed by system() rather than running tool.
  command_only = FALSE
)
```

**Arguments**

- `inputs`  
  Input raster files.
- `output`  
  Output raster file.
- `out_html`  
  Output HTML report file.
- `start_clusters`  
  Initial number of clusters.
- `merge_dist`  
  Cluster merger distance.
- `max_iterations`  
  Maximum number of iterations.
- `class_change`  
  Minimum percent of cells changed between iterations before completion.
- `wd`  
  Changes the working directory.
- `verbose_mode`  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
Modify no data value

Description
Converts nodata values in a raster to zero.

Usage
wbt_modify_no_data_value(
    input,
    new_value = "-32768.0",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments
input Input raster file.
new_value New NoData value.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.
wbt_modulo  Modulo

Description

Performs a modulo operation on two rasters or a raster and a constant value.

Usage

wbt_modulo(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input1    Input raster file or constant value.
input2    Input raster file or constant value.
output    Output raster file.
wd        Changes the working directory.
verbose_mode    Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters    Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only    Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
Mosaics two or more images together.

Usage

```r
wbt_mosaic(
    output,
    inputs = NULL,
    method = "nn",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

Arguments

- **output**: Output raster file.
- **inputs**: Input raster files.
- **method**: Resampling method; options include 'nn' (nearest neighbour), 'bilinear', and 'cc' (cubic convolution).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_mosaic_with_feathering

Mosaic with feathering

Description

Mosaics two images together using a feathering technique in overlapping areas to reduce edge-effects.

Usage

```r
wbt_mosaic_with_feathering(
  input1,
  input2,
  output,
  method = "cc",
  weight = 4,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input1**: Input raster file to modify.
- **input2**: Input reference raster file.
- **output**: Output raster file.
- **method**: Resampling method; options include 'nn' (nearest neighbour), 'bilinear', and 'cc' (cubic convolution).
- **weight**: .
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
Description

Calculates a multi-direction hillshade raster from an input DEM.

Usage

```r
wbt_multidirectional_hillshade(
  dem,
  output,
  altitude = 45,
  zfactor = NULL,
  full_mode = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

dem Input raster DEM file.
output Output raster file.
altitude Illumination source altitude in degrees.
zfactor Optional multiplier for when the vertical and horizontal units are not the same.
full_mode Optional flag indicating whether to use full 360-degrees of illumination sources.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
Description

Performs a multiplication operation on two rasters or a raster and a constant value.

Usage

wbt_multiply(
    input1,
    input2,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input1  Input raster file or constant value.
input2  Input raster file or constant value.
output  Output raster file.
wd      Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
**wbt_multiscale_elevation_percentile**

*Multiscale elevation percentile*

**Description**

Calculates surface roughness over a range of spatial scales.

**Usage**

```r
wbt_multiscale_elevation_percentile(
  dem,
  out_mag,
  out_scale,
  sig_digits = 3,
  min_scale = 4,
  step = 1,
  num_steps = 10,
  step_nonlinearity = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>dem</code></td>
<td>Input raster DEM file.</td>
</tr>
<tr>
<td><code>out_mag</code></td>
<td>Output raster roughness magnitude file.</td>
</tr>
<tr>
<td><code>out_scale</code></td>
<td>Output raster roughness scale file.</td>
</tr>
<tr>
<td><code>sig_digits</code></td>
<td>Number of significant digits.</td>
</tr>
<tr>
<td><code>min_scale</code></td>
<td>Minimum search neighbourhood radius in grid cells.</td>
</tr>
<tr>
<td><code>step</code></td>
<td>Step size as any positive non-zero integer.</td>
</tr>
<tr>
<td><code>num_steps</code></td>
<td>Number of steps.</td>
</tr>
<tr>
<td><code>step_nonlinearity</code></td>
<td>Step nonlinearity factor (1.0-2.0 is typical).</td>
</tr>
<tr>
<td><code>wd</code></td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td><code>verbose_mode</code></td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td><code>compress_rasters</code></td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td><code>command_only</code></td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>
wbt_multiscale_roughness

Multiscale roughness

Description

Calculates surface roughness over a range of spatial scales.

Usage

```r
wbt_multiscale_roughness(
    dem,
    out_mag,
    out_scale,
    max_scale,
    min_scale = 1,
    step = 1,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

Arguments

- **dem**: Input raster DEM file.
- **out_mag**: Output raster roughness magnitude file.
- **out_scale**: Output raster roughness scale file.
- **max_scale**: Maximum search neighbourhood radius in grid cells.
- **min_scale**: Minimum search neighbourhood radius in grid cells.
- **step**: Step size as any positive non-zero integer.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
Multiscale roughness signature

Description

Calculates the surface roughness for points over a range of spatial scales.

Usage

```
wbt_multiscale_roughness_signature(
    dem,  # Input raster DEM file.
    points,  # Input vector points file.
    output,  # Output HTML file.
    max_scale,  # Maximum search neighbourhood radius in grid cells.
    min_scale = 1,  # Minimum search neighbourhood radius in grid cells.
    step = 1,  # Step size as any positive non-zero integer.
    wd = NULL,  # Changes the working directory.
    verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
    compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
    command_only = FALSE  # Return command that would be executed by system() rather than running tool.
)
```

Arguments

- **dem**: Input raster DEM file.
- **points**: Input vector points file.
- **output**: Output HTML file.
- **max_scale**: Maximum search neighbourhood radius in grid cells.
- **min_scale** (default: 1): Minimum search neighbourhood radius in grid cells.
- **step** (default: 1): Step size as any positive non-zero integer.
- **wd**: Changes the working directory.
- **verbose_mode** (default: FALSE): Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters** (default: FALSE): Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only** (default: FALSE): Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
Description

Calculates surface roughness over a range of spatial scales.

Usage

wbt_multiscale_std_dev_normals(
  dem,
  out_mag,  
  out_scale, 
  min_scale = 1,
  step = 1,
  num_steps = 10,
  step_nonlinearity = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

dem  Input raster DEM file.
out_mag Output raster roughness magnitude file.
out_scale Output raster roughness scale file.
min_scale Minimum search neighbourhood radius in grid cells.
step  Step size as any positive non-zero integer.
um_steps Number of steps.
step_nonlinearity Step nonlinearity factor (1.0-2.0 is typical).
wd  Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
Multiscale std dev normals signature

Description
Calculates the surface roughness for points over a range of spatial scales.

Usage

```r
wbt_multiscale_std_dev_normals_signature(
  dem,
  points,
  output,
  min_scale = 1,
  step = 1,
  num_steps = 10,
  step_nonlinearity = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments
- `points`: Input vector points file.
- `output`: Output HTML file.
- `min_scale`: Minimum search neighbourhood radius in grid cells.
- `step`: Step size as any positive non-zero integer.
- `num_steps`: Number of steps.
- `step_nonlinearity`: Step nonlinearity factor (1.0-2.0 is typical).
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value
Returns the tool text outputs.
wbt_multiscale_topographic_position_image

Multiscale topographic position image

Description

Creates a multiscale topographic position image from three DEVmax rasters of differing spatial scale ranges.

Usage

wbt_multiscale_topographic_position_image(
    local,
    meso,
    broad,
    output,
    lightness = 1.2,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

local          Input local-scale topographic position (DEVmax) raster file.
meso           Input meso-scale topographic position (DEVmax) raster file.
broad          Input broad-scale topographic position (DEVmax) raster file.
output         Output raster file.
lightness      Image lightness value (default is 1.2).
wd             Changes the working directory.
verbose_mode   Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only   Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
Multi part to single part

Description

Converts a vector file containing multi-part features into a vector containing only single-part features.

Usage

wbt_multi_part_to_single_part(
  input,
  output,
  exclude_holes = TRUE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input Input vector line or polygon file.
output Output vector line or polygon file.
exclude_holes Exclude hole parts from the feature splitting? (holes will continue to belong to their features in output.).
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
**wbt_narrowness_index**  
**Narrowness index**

**Description**
Calculates the narrowness of raster polygons.

**Usage**
```r
wbt_narrowness_index(
  input, output, wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**
- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**
Returns the tool text outputs.

**Description**
Creates a raster grid based on Sibson’s natural neighbour method.
wbt_nearest_neighbour_gridding

Usage

wbt_natural_neighbour_interpolation(
    input,
    output,
    field = NULL,
    use_z = FALSE,
    cell_size = NULL,
    base = NULL,
    clip = TRUE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input Input vector points file.
output Output raster file.
field Input field name in attribute table.
use_z Use the 'z' dimension of the Shapefile's geometry instead of an attribute field?
cell_size Optionally specified cell size of output raster. Not used when base raster is specified.
base Optionally specified input base raster file. Not used when a cell size is specified.
clip Clip the data to the convex hull of the points?.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_nearest_neighbour_gridding

Nearest neighbour gridding

Description

Creates a raster grid based on a set of vector points and assigns grid values using the nearest neighbour.
Usage

wbt_nearest_neighbour_gridding(
    input,
    field,
    output,
    use_z = FALSE,
    cell_size = NULL,
    base = NULL,
    max_dist = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input  Input vector Points file.
field  Input field name in attribute table.
output Output raster file.
use_z  Use z-coordinate instead of field?.
cell_size Optionally specified cell size of output raster. Not used when base raster is specified.
base  Optionally specified input base raster file. Not used when a cell size is specified.
max_dist Maximum search distance (optional).
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

wbt_negate  Negate

Description

Changes the sign of values in a raster or the 0-1 values of a Boolean raster.
Usage

wbt_negate(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input Input raster file.
output Output raster file.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

New raster from base

Creates a new raster using a base image.
Arguments

- **base**: Input base raster file.
- **output**: Output raster file.
- **value**: Constant value to fill raster with; either 'nodata' or numeric value.
- **data_type**: Output raster data type; options include 'double' (64-bit), 'float' (32-bit), and 'integer' (signed 16-bit) (default is 'float').
- **cell_size**: Optionally specified cell size of output raster. Not used when base raster is specified.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

```
wd_normalized_difference_index

Normalized difference index
```

Description

Calculate a normalized-difference index (NDI) from two bands of multispectral image data.

Usage

```r
wd_normalized_difference_index(
  input1, input2, output, clip = 0,
  correction = 0, wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE, command_only = FALSE
)
```
**wbt_normal_vectors**

**Arguments**

- **input1**: Input image 1 (e.g. near-infrared band).
- **input2**: Input image 2 (e.g. red band).
- **output**: Output raster file.
- **clip**: Optional amount to clip the distribution tails by, in percent.
- **correction**: Optional adjustment value (e.g. 1, or 0.16 for the optimal soil adjusted vegetation index, OSAVI).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_normal_vectors**  
**Normal vectors**

**Description**

Calculates normal vectors for points within a LAS file and stores these data (XYZ vector components) in the RGB field.

**Usage**

```r
wbt_normal_vectors(
  input, 
  output, 
  radius = 1, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE 
)
```
Arguments

input  Input LiDAR file.
output Output LiDAR file.
radius Search Radius.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

**Description**

Performs a logical NOT operator on two Boolean raster images.

**Usage**

```r
wbt_not(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input1 Input raster file.
input2 Input raster file.
output Output raster file.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.
Value

Returns the tool text outputs.

---

Description

Performs a not-equal-to comparison operation on two rasters or a raster and a constant value.

Usage

```r
wbt_not_equal_to(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `input1`: Input raster file or constant value.
- `input2`: Input raster file or constant value.
- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
Description

Calculates the number of downslope neighbours to each grid cell in a DEM.

Usage

```r
wbt_num_downslope_neighbours(
  dem,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

dem Input raster DEM file.
output Output raster file.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Computes the number of inflowing neighbours to each cell in an input DEM based on the D8 algorithm.
wbt_num_upslope_neighbours

Usage

wbt_num_inflowing_neighbours(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

dem                      Input raster DEM file.
output                   Output raster file.
wd                       Changes the working directory.
verbose_mode             Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters         Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only             Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_num_upslope_neighbours

Num upslope neighbours

Description

Calculates the number of upslope neighbours to each grid cell in a DEM.

Usage

wbt_num_upslope_neighbours(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
Arguments

- **dem**  
  Input raster DEM file.
- **output**  
  Output raster file.
- **wd**  
  Changes the working directory.
- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**  
  Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_olympic_filter**  
*Olympic filter*

Description

Performs an olympic smoothing filter on an image.

Usage

```r
wbt_olympic_filter(
  input,
  output,
  filterx = 11,
  filtery = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**  
  Input raster file.
- **output**  
  Output raster file.
- **filterx**  
  Size of the filter kernel in the x-direction.
- **filtery**  
  Size of the filter kernel in the y-direction.
- **wd**  
  Changes the working directory.
- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
wbt_opening

compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

wbt_opening (Opening)

Description
An opening is a mathematical morphology operation involving a dilation (max filter) of an erosion (min filter) set.

Usage
wbt_opening(
    input,
    output,
    filterx = 11,
    filtery = 11,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments
input        Input raster file.
output       Output raster file.
filterx      Size of the filter kernel in the x-direction.
filtery      Size of the filter kernel in the y-direction.
wd           Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.
**Description**

This tool calculates the topographic openness index from an input DEM.

**Usage**

```r
wbt_openness(
  input, 
  pos_output, 
  neg_output, 
  dist = 20, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)
```

**Arguments**

- **input** Name of the input raster DEM file.
- **pos_output** Name of the positive openness output raster file.
- **neg_output** Name of the negative openness output raster file.
- **dist** Search distance, in grid cells.
- **wd** Changes the working directory.
- **verbose_mode** Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters** Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only** Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**Description**

Performs a logical OR operator on two Boolean raster images.

**Usage**

```r
wbt_or(
    input1,  
    input2,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE
)
```

**Arguments**

- `input1` Input raster file.
- `input2` Input raster file.
- `output` Output raster file.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_paired_sample_t_test

*Paired sample t test*

**Description**

Performs a 2-sample K-S test for significant differences on two input rasters.

**Usage**

```r
wbt_paired_sample_t_test(
  input1,
  input2,
  output,
  num_samples = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input1**: First input raster file.
- **input2**: Second input raster file.
- **output**: Output HTML file.
- **num_samples**: Number of samples. Leave blank to use whole image.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_panchromatic_sharpening

Panchromatic sharpening

Description

Increases the spatial resolution of image data by combining multispectral bands with panchromatic data.

Usage

wbt_panchromatic_sharpening(
    pan, output,
    red = NULL, green = NULL, blue = NULL,
    composite = NULL, method = "brovey",
    wd = NULL, verbose_mode = FALSE, compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

    pan Input panchromatic band file.
    output Output colour composite file.
    red Input red band image file. Optionally specified if colour-composite not specified.
    green Input green band image file. Optionally specified if colour-composite not specified.
    blue Input blue band image file. Optionally specified if colour-composite not specified.
    composite Input colour-composite image file. Only used if individual bands are not specified.
    method Options include 'brovey' (default) and 'ihs'.
    wd Changes the working directory.
    verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
    compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
    command_only Return command that would be executed by system() rather than running tool.
wbt_parallelepiped_classification

Parallelepiped classification

Description

Performs a supervised parallelepiped classification using training site polygons and multi-spectral images.

Usage

```r
wbt_parallelepiped_classification(
  inputs,
  polys,
  field,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **inputs**: Name of the input band images.
- **polys**: Name of the input training site polygons shapefile.
- **field**: Name of the attribute containing class name data.
- **output**: Name of the output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
Description

Calculates the orientation of vector polygons.

Usage

```r
wbt_patch_orientation(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input vector polygon file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Classifies hillslope zones based on slope, profile curvature, and plan curvature.
Usage

\[
\text{wbt\_pennock\_landform\_class(}
\begin{align*}
\text{dem}, \\
\text{output}, \\
\text{slope} & = 3, \\
\text{prof} & = 0.1, \\
\text{plan} & = 0, \\
\text{zfactor} & = \text{NULL}, \\
\text{wd} & = \text{NULL}, \\
\text{verbose\_mode} & = \text{FALSE}, \\
\text{compress\_rasters} & = \text{FALSE}, \\
\text{command\_only} & = \text{FALSE}
\end{align*}
\]

Arguments

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **slope**: Slope threshold value, in degrees (default is 3.0).
- **prof**: Profile curvature threshold value (default is 0.1).
- **plan**: Plan curvature threshold value (default is 0.0).
- **zfactor**: Optional multiplier for when the vertical and horizontal units are not the same.
- **wd**: Changes the working directory.
- **verbose\_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress\_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command\_only**: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

### Description

Performs a percentage linear contrast stretch on input images.
Usage

```r
wbt_percentage_contrast_stretch(
    input,
    output,
    clip = 1,
    tail = "both",
    num_tones = 256,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **clip**: Optional amount to clip the distribution tails by, in percent.
- **tail**: Specified which tails to clip; options include 'upper', 'lower', and 'both' (default is 'both').
- **num_tones**: Number of tones in the output image.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Performs a percentile filter on an input image.
Usage

```r
wbt_percentile_filter(
  input, output,
  filterx = 11,
  filtery = 11,
  sig_digits = 2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **filterx**: Size of the filter kernel in the x-direction.
- **filtery**: Size of the filter kernel in the y-direction.
- **sig_digits**: Number of significant digits.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Calculates percent of elevation range from a DEM.
**wbt_percent_equal_to**

**Usage**

```r
wbt_percent_elev_range(
  dem,  
  output,  
  filterx = 3,  
  filtery = 3,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)
```

**Arguments**

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **filterx**: Size of the filter kernel in the x-direction.
- **filtery**: Size of the filter kernel in the y-direction.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_percent_equal_to  Percent equal to**

**Description**

Calculates the percentage of a raster stack that have cell values equal to an input on a cell-by-cell basis.

**Usage**

```r
wbt_percent_equal_to(
  inputs,  
  comparison,  
  output,  
  wd = NULL,
)```
wbt_percent_greater_than

Arguments

- **inputs**: Input raster files.
- **comparison**: Input comparison raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Calculates the percentage of a raster stack that have cell values greater than an input on a cell-by-cell basis.

Usage

```r
wbt_percent_greater_than(
  inputs,
  comparison,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
Arguments

inputs: Input raster files.
comparison: Input comparison raster file.
output: Output raster file.
wd: Changes the working directory.
verbose_mode: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_percent_less_than**  
*Percent less than*

Description

Calculates the percentage of a raster stack that have cell values less than an input on a cell-by-cell basis.

Usage

```r
wbt_percent_less_than(  
  inputs,  
  comparison,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

Arguments

inputs: Input raster files.
comparison: Input comparison raster file.
output: Output raster file.
wd: Changes the working directory.
verbose_mode: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
`wbt_perimeter_area_ratio`

Perimeter area ratio

**Description**

Calculates the perimeter-area ratio of vector polygons.

**Usage**

```r
wbt_perimeter_area_ratio(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input vector polygon file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_phi_coefficient

**Description**

This tool performs a binary classification accuracy assessment.

**Usage**

```r
wbt_phi_coefficient(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input1` Name of the first input raster image file.
- `input2` Name of the second input raster image file.
- `output` Name of the output HTML file.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_pick_from_list  Pick from list

Description

Outputs the value from a raster stack specified by a position raster.

Usage

wbt_pick_from_list(
  inputs,
  pos_input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

inputs  Input raster files.
pos_input  Input position raster file.
output  Output raster file.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
wbt_plan_curvature

Plan curvature

Description

Calculates a plan (contour) curvature raster from an input DEM.

Usage

```r
wbt_plan_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **dem**  
  Input raster DEM file.
- **output**  
  Output raster file.
- **log**  
  Display output values using a log-scale.
- **zfactor**  
  Optional multiplier for when the vertical and horizontal units are not the same.
- **wd**  
  Changes the working directory.
- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**  
  Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**wbt_polygonize**  
*Polygonize*

**Description**

Creates a polygon layer from two or more intersecting line features contained in one or more input vector line files.

**Usage**

```r
wbt_polygonize(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **inputs**: Input vector polyline file.
- **output**: Output vector polygon file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_polygons_to_lines**  
*Polygons to lines*

**Description**

Converts vector polygons to polylines.
wbt_polygon_area

Usage

wbt_polygons_to_lines(
    input, output, wd = NULL, verbose_mode = FALSE, compress_rasters = FALSE, command_only = FALSE
)

Arguments

input       Input vector polygon file.
output      Output vector lines file.
wd          Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_polygon_area   Polygon area

Description

Calculates the area of vector polygons.

Usage

wbt_polygon_area(
    input, wd = NULL, verbose_mode = FALSE, compress_rasters = FALSE, command_only = FALSE
)
```
Arguments

input  Input vector polygon file.
wd     Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
```

**wbt_polygon_long_axis**  *Polygon long axis*

**Description**

This tool can be used to map the long axis of polygon features.

**Usage**

```r
wbt_polygon_long_axis(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

```plaintext
input  Input vector polygons file.
output  Output vector polyline file.
wd     Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by `system()` rather than running tool.
```

**Value**

Returns the tool text outputs.
**wbt_polygon_perimeter**  
*Polygon perimeter*

**Description**

Calculates the perimeter of vector polygons.

**Usage**

```r
wbt_polygon_perimeter(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input vector polygon file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_polygon_short_axis**  
*Polygon short axis*

**Description**

This tool can be used to map the short axis of polygon features.
Usage

```r
wbt_polygon_short_axis(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input vector polygons file.
- **output**: Output vector polyline file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

**Power**

 Raises the values in grid cells of one rasters, or a constant value, by values in another raster or constant value.

Usage

```r
wbt_power(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
Arguments

input1  Input raster file or constant value.
input2  Input raster file or constant value.
output  Output raster file.
wd      Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Performs a Prewitt edge-detection filter on an image.

Usage

```r
wbt_prewitt_filter(
  input,
  output,
  clip = 0,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

input  Input raster file.
output  Output raster file.
clip   Optional amount to clip the distribution tails by, in percent.
wd     Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.
Principal component analysis

Performs a principal component analysis (PCA) on a multi-spectral dataset.

Usage

```r
wbt_principal_component_analysis(
  inputs,
  output,
  num_comp = NULL,
  standardized = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **inputs**: Input raster files.
- **output**: Output HTML report file.
- **num_comp**: Number of component images to output; <= to num. input images.
- **standardized**: Perform standardized PCA?.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_print_geo_tiff_tags

Print geo tiff tags

Description

Prints the tags within a GeoTIFF.

Usage

\[
\text{wbt_print_geo_tiff_tags(}
\text{\hspace{1em}}\text{input,}
\text{\hspace{1em}}\text{wd = NULL,}
\text{\hspace{1em}}\text{verbose_mode = FALSE,}
\text{\hspace{1em}}\text{compress_rasters = FALSE,}
\text{\hspace{1em}}\text{command_only = FALSE}
\text{)}
\]

Arguments

input  \hspace{1em} Input GeoTIFF file.
wd  \hspace{1em} Changes the working directory.
verbose_mode  \hspace{1em} Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  \hspace{1em} Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  \hspace{1em} Return command that would be executed by \texttt{system()} rather than running tool.

Value

Returns the tool text outputs.

wbt_profile

Profile

Description

Plots profiles from digital surface models.
Usage

```r
wbt_profile(
  lines,
  surface,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `lines`: Input vector line file.
- `output`: Output HTML file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_profile_curvature**  Profile curvature

Description

Calculates a profile curvature raster from an input DEM.

Usage

```r
wbt_profile_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
Arguments

dem Input raster DEM file.
output Output raster file.
log Display output values using a log-scale.
zfactor Optional multiplier for when the vertical and horizontal units are not the same.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_qin_flow_accumulation**

*Qin flow accumulation*

---

Description

This tool calculates Qin et al. (2007) flow accumulation.

Usage

```r
wbt_qin_flow_accumulation(
  dem,  
  output, 
  out_type = "specific contributing area", 
  exponent = 10, 
  max_slope = 45, 
  threshold = NULL, 
  log = FALSE, 
  clip = FALSE, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)
```
### Arguments

- **dem**
  Name of the input DEM raster file; must be depressionless.

- **output**
  Name of the output raster file.

- **out_type**
  Output type; one of 'cells', 'specific contributing area' (default), and 'catchment area'.

- **exponent**
  Optional upper-bound exponent parameter; default is 10.0.

- **max_slope**
  Optional upper-bound slope parameter, in degrees (0-90); default is 45.0.

- **threshold**
  Optional convergence threshold parameter, in grid cells; default is infinity.

- **log**
  Log-transform the output values?.

- **clip**
  Optional flag to request clipping the display max by 1 percent.

- **wd**
  Changes the working directory.

- **verbose_mode**
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

- **compress_rasters**
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

- **command_only**
  Return command that would be executed by `system()` rather than running tool.

### Value

Returns the tool text outputs.

---

### Description

Transforms raster values into quantiles.

### Usage

```r
wbt_quantiles(
  input, output, num_quantiles = 5, wd = NULL,
  verbose_mode = FALSE, compress_rasters = FALSE,
  command_only = FALSE
)
```
Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **num_quantiles**: Number of quantiles.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

```
wb_quinn_flow_accumulation

Quinn flow accumulation
```

Description

This tool calculates Quinn et al. (1995) flow accumulation.

Usage

```
wb_quinn_flow_accumulation(
  dem,
  output,
  out_type = "specific contributing area",
  exponent = 1,
  threshold = NULL,
  log = FALSE,
  clip = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
wbt_radial_basis_function_interpolation

Arguments

dem Name of the input DEM raster file; must be depressionless.
output Name of the output raster file.
out_type Output type; one of 'cells', 'specific contributing area' (default), and 'catchment area'.
exponent Optional exponent parameter; default is 1.0.
threshold Optional convergence threshold parameter, in grid cells; default is infinity.
log Log-transform the output values?.
clip Optional flag to request clipping the display max by 1 percent.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_radial_basis_function_interpolation

Radial basis function interpolation

Description

Interpolates vector points into a raster surface using a radial basis function scheme.

Usage

wbt_radial_basis_function_interpolation(input, field, output, use_z = FALSE, radius = NULL, min_points = NULL, func_type = "ThinPlateSpline", poly_order = "none", weight = 0.1, cell_size = NULL, base = NULL, wd = NULL,
Arguments

- **input**: Input vector points file.
- **field**: Input field name in attribute table.
- **output**: Output raster file.
- **use_z**: Use z-coordinate instead of field?.
- **radius**: Search Radius (in map units).
- **min_points**: Minimum number of points.
- **func_type**: Radial basis function type; options are 'ThinPlateSpline' (default), 'PolyHarmonic', 'Gaussian', 'MultiQuadric', 'InverseMultiQuadric'.
- **poly_order**: Polynomial order; options are 'none' (default), 'constant', 'affine'.
- **weight**: Weight parameter used in basis function.
- **cell_size**: Optionally specified cell size of output raster. Not used when base raster is specified.
- **base**: Optionally specified input base raster file. Not used when a cell size is specified.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_radius_of_gyration**

*Radius of gyration*

Description

Calculates the distance of cells from their polygon’s centroid.
**Usage**

```r
wbt_radius_of_gyration(
  input,
  output,
  text_output = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **text_output**: Optional text output.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_raise_walls**

**Raise walls**

**Description**

Raises walls in a DEM along a line or around a polygon, e.g., a watershed.

**Usage**

```r
wbt_raise_walls(
  input,
  dem,
  output,
  breach = NULL,
  height = 100,
  wd = NULL,
  verbose_mode = FALSE,
```
wbt_random_field

Arguments

- **input**: Input vector lines or polygons file.
- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **breach**: Optional input vector breach lines.
- **height**: Wall height.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Creates an image containing random values.

Usage

```r
wbt_random_field(
  base, output, wd = NULL, verbose_mode = FALSE, compress_rasters = FALSE, command_only = FALSE
)
```
Arguments

- **base**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_random_forest_classification**

*Random forest classification*

Description

Performs a supervised random forest classification using training site polygons/points and predictor rasters.

Usage

```r
wbt_random_forest_classification(
  inputs,
  training,
  field,
  output = NULL,
  split_criterion = "Gini",
  n_trees = 500,
  min_samples_leaf = 1,
  min_samples_split = 2,
  test_proportion = 0.2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
Arguments

- **inputs**: Names of the input predictor rasters.
- **training**: Name of the input training site polygons/points shapefile.
- **field**: Name of the attribute containing class data.
- **output**: Name of the output raster file.
- **split_criterion**: Split criterion to use when building a tree. Options include 'Gini', 'Entropy', and 'ClassificationError'.
- **n_trees**: The number of trees in the forest.
- **min_samples_leaf**: The minimum number of samples required to be at a leaf node.
- **min_samples_split**: The minimum number of samples required to split an internal node.
- **test_proportion**: The proportion of the dataset to include in the test split; default is 0.2.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Performs a random forest regression analysis using training site data and predictor rasters.

Usage

```r
wbt_random_forest_regression(
  inputs,  # Names of the input predictor rasters.
  training,  # Name of the input training site polygons/points shapefile.
  field,  # Name of the attribute containing class data.
  output = NULL,  # Name of the output raster file.
  n_trees = 100,  # The number of trees in the forest.
  split_criterion = 'Gini',  # Split criterion to use when building a tree. Options include 'Gini', 'Entropy', and 'ClassificationError'.
  min_samples_leaf = 1,  # The minimum number of samples required to be at a leaf node.
  min_samples_split = 2,  # The minimum number of samples required to split an internal node.
  test_proportion = 0.2,  # The proportion of the dataset to include in the test split; default is 0.2.
  wd = getwd(),  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE)  # Return command that would be executed by `system()` rather than running tool.
```
min_samples_leaf = 1,
min_samples_split = 2,
test_proportion = 0.2,
wd = NULL,
verbose_mode = FALSE,
compress_rasters = FALSE,
command_only = FALSE
)

Arguments

inputs  Names of the input predictor rasters.
training Name of the input training site points shapefile.
field Name of the attribute containing response variable name data.
output Name of the output raster file. This parameter is optional. When unspecified, the tool will only build the model. When specified, the tool will use the built model and predictor rasters to perform a spatial prediction.
n_trees The number of trees in the forest.
min_samples_leaf The minimum number of samples required to be at a leaf node.
min_samples_split The minimum number of samples required to split an internal node.
test_proportion The proportion of the dataset to include in the test split; default is 0.2.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_random_sample**  **Random sample**

**Description**

Creates an image containing randomly located sample grid cells with unique IDs.
Usage

```r
wbt_random_sample(
    base,
    output,
    num_samples = 1000,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

Arguments

- `output`: Output raster file.
- `num_samples`: Number of samples.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Assigns each cell in the output grid the range of values in a moving window centred on each grid cell in the input raster.

Usage

```r
wbt_range_filter(
    input,
    output,
    filterx = 11,
    filtery = 11,
    wd = NULL,
    verbose_mode = FALSE,
)```
wbt_rasterize_streams

compress_rasters = FALSE,
command_only = FALSE

Arguments

input      Input raster file.
output     Output raster file.
filterx    Size of the filter kernel in the x-direction.
filtery    Size of the filter kernel in the y-direction.
wd         Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_rasterize_streams  Rasterize streams

Description

Rasterizes vector streams based on Lindsay (2016) method.

Usage

wbt_rasterize_streams(
    streams,
    base,
    output,
    nodata = TRUE,
    feature_id = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
wbt_raster_area

Arguments

streams Input vector streams file.
base Input base raster file.
output Output raster file.
nodata Use NoData value for background?.
feature_id Use feature number as output value?.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_raster_area  Raster area

Description

Calculates the area of polygons or classes within a raster image.

Usage

wbt_raster_area(
  input,
  output = NULL,
  out_text = FALSE,
  units = "grid cells",
  zero_back = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **out_text**: Would you like to output polygon areas to text?
- **units**: Area units; options include 'grid cells' and 'map units'.
- **zero_back**: Flag indicating whether zero values should be treated as a background.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_raster_calculator**  
*Raster calculator*

Description

This tool performs a complex mathematical operations on one or more input raster images on a cell-to-cell basis.

Usage

```r
wbt_raster_calculator(
  output,
  statement = "",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **output**: Name of the output raster file.
- **statement**: Statement e.g. \( \cos("raster1") \times 35.0 + \)"raster2". This statement must be a valid Rust statement.
- **wd**: Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

wbt_raster_cell_assignment  

**Raster cell assignment**

**Description**

Assign row or column number to cells.

**Usage**

```r
wbt_raster_cell_assignment(  
  input,  
  output,  
  assign = "column",  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **assign**: Which variable would you like to assign to grid cells? Options include 'column', 'row', 'x', and 'y'.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.
Value

Returns the tool text outputs.

---

**wbt_raster_histogram**  
*Raster histogram*

---

**Description**

Creates a histogram from raster values.

**Usage**

```r
wbt_raster_histogram(
  input,  
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input`  
  Input raster file.
- `output`  
  Output HTML file (default name will be based on input file if unspecified).
- `wd`  
  Changes the working directory.
- `verbose_mode`  
  Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`  
  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
Description

Calculates the perimeters of polygons or classes within a raster image.

Usage

```r
wbt_raster_perimeter(
  input,
  output = NULL,
  out_text = FALSE,
  units = "grid cells",
  zero_back = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `input` | Input raster file.
- `output` | Output raster file.
- `out_text` | Would you like to output polygon areas to text?.
- `units` | Area units; options include 'grid cells' and 'map units'.
- `zero_back` | Flag indicating whether zero values should be treated as a background.
- `wd` | Changes the working directory.
- `verbose_mode` | Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters` | Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` | Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**wbt_raster_streams_to_vector**

*Raster streams to vector*

**Description**

Converts a raster stream file into a vector file.

**Usage**

```r
wbt_raster_streams_to_vector(
  streams,
  d8_pntr,
  output,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `streams` Input raster streams file.
- `d8_pntr` Input raster D8 pointer file.
- `output` Output vector file.
- `esri_pntr` D8 pointer uses the ESRI style scheme.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_raster_summary_stats**

*Raster summary stats*

**Description**

Measures a raster's min, max, average, standard deviation, num. non-nodata cells, and total.

**Usage**

```r
wbt_raster_summary_stats(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input`  
  Input raster file.
- `wd`  
  Changes the working directory.
- `verbose_mode`  
  Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`  
  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_raster_to_vector_lines**

*Raster to vector lines*

**Description**

Converts a raster lines features into a vector of the POLYLINE shapetype.
Usage

```r
wbt_raster_to_vector_lines(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `input`: Input raster lines file.
- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If `verbose_mode` is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

**Description**

Converts a raster dataset to a vector of the POINT shapetype.

Usage

```r
wbt_raster_to_vector_points(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
**Arguments**

- **input**: Input raster file.
- **output**: Output vector points file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**Description**

Converts a raster dataset to a vector of the POLYGON shapetype.

**Usage**

```r
wbt_raster_to_vector_polygons(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output vector polygons file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.
Value

Returns the tool text outputs.


Description

Returns the reciprocal (i.e. $1 / z$) of values in a raster.

Usage

```r
wbt_reciprocal(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `input` Input raster file.
- `output` Output raster file.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_reclass  

Reclass

Description
Reclassifies the values in a raster image.

Usage
wbt_reclass(
  input,  
  output,  
  reclass_vals,  
  assign_mode = FALSE,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)

Arguments
input  Input raster file.
output  Output raster file.
reclass_vals  Reclassification triplet values (new value; from value; to less than), e.g. '0.0;0.0;1.0;1.0;1.0;2.0'.
assign_mode  Optional Boolean flag indicating whether to operate in assign mode, reclass_vals values are interpreted as new value; old value pairs.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.
Reclass equal interval

Description

Reclassifies the values in a raster image based on equal-ranges.

Usage

```r
wbt_reclass_equal_interval(
  input,
  output,
  interval = 10,
  start_val = NULL,
  end_val = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **interval**: Class interval size.
- **start_val**: Optional starting value (default is input minimum value).
- **end_val**: Optional ending value (default is input maximum value).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
Reclass from file

Description
Reclassifies the values in a raster image using reclass ranges in a text file.

Usage
```r
wbt_reclass_from_file(
  input,
  reclass_file,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments
- `reclass_file`: Input text file containing reclass ranges.
- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value
Returns the tool text outputs.
wbt_reconcile_multiple_headers

Reconcile multiple headers

Description

This tool adjusts the crop yield values for data sets collected with multiple headers or combines.

Usage

```
wbt_reconcile_multiple_headers(
  input,
  region_field,
  yield_field,
  output,
  radius = NULL,
  min_yield = NULL,
  max_yield = NULL,
  mean_tonnage = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Name of the input points shapefile.</td>
</tr>
<tr>
<td>region_field</td>
<td>Name of the attribute containing region data.</td>
</tr>
<tr>
<td>yield_field</td>
<td>Name of the attribute containing yield data.</td>
</tr>
<tr>
<td>output</td>
<td>Name of the output points shapefile.</td>
</tr>
<tr>
<td>radius</td>
<td>Optional search radius, in metres. Only specify this value if you want to calculate locally normalized yield.</td>
</tr>
<tr>
<td>min_yield</td>
<td>Minimum yield value in output.</td>
</tr>
<tr>
<td>max_yield</td>
<td>Maximum yield value in output.</td>
</tr>
<tr>
<td>mean_tonnage</td>
<td>Use this optional parameter to force the output to have a certain overall average tonnage.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>
### Description

This tool can be used to approximate the harvester pass lines from yield points.

### Usage

```r
wbt_recreate_pass_lines(
  input,  # Name of the input points shapefile.
  yield_field_name,  # Name of the attribute containing yield data.
  output_lines,  # Name of the output pass lines shapefile.
  output_points,  # Name of the output points shapefile.
  max_change_in_heading = 25,  # Max change in heading.
  ignore_zeros = FALSE,  # Ignore zero-valued yield points?.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE  # Return command that would be executed by `system()` rather than running tool.
)
```

### Arguments

- **Input**: Name of the input points shapefile.
- **Yield field name**: Name of the attribute containing yield data.
- **Output lines**: Name of the output pass lines shapefile.
- **Output points**: Name of the output points shapefile.
- **Max change in heading**: Max change in heading.
- **Ignore zeros**: Ignore zero-valued yield points?.
- **Working directory**: Changes the working directory.
- **Verbose mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **Compress rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **Command only**: Return command that would be executed by `system()` rather than running tool.
wbt_reinitialize_attribute_table

*Reinitialize attribute table*

**Description**

Reinitializes a vector’s attribute table deleting all fields but the feature ID (FID).

**Usage**

```r
wbt_reinitialize_attribute_table(
    input,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

- `input` Input vector file.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_related_circumscribing_circle**

*Related circumscribing circle*

**Description**

Calculates the related circumscribing circle of vector polygons.

**Usage**

```r
wbt_related_circumscribing_circle(
  input, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)
```

**Arguments**

- **input**
  - Input vector polygon file.
- **wd**
  - Changes the working directory.
- **verbose_mode**
  - Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**
  - Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**
  - Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_relative_aspect**

*Relative aspect*

**Description**

Calculates relative aspect (relative to a user-specified direction) from an input DEM.
Usage

wbt_relative_aspect(
    dem,
    output,
    azimuth = 0,
    zfactor = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

dem Input raster DEM file.
output Output raster file.
azimuth Illumination source azimuth.
zfactor Optional multiplier for when the vertical and horizontal units are not the same.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Calculates the relative topographic position index from a DEM.

Usage

wbt_relative_topographic_position(
    dem,
    output,
    filterx = 11,
    filtery = 11,
wbt_remove_field_edge_points

wd = NULL,
verbose_mode = FALSE,
compress_rasters = FALSE,
command_only = FALSE
)

Arguments

dem Input raster DEM file.
output Output raster file.
filterx Size of the filter kernel in the x-direction.
filtery Size of the filter kernel in the y-direction.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_remove_field_edge_points

Remove field edge points

Description

This tool can be used to remove, or flag, most of the points along the edges from a crop yield data set.

Usage

wbt_remove_field_edge_points(
  input,
  output,
  dist = NULL,
  max_change_in_heading = 25,
  flag_edges = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
Arguments

- **input**: Name of the input points shapefile.
- **output**: Name of the output points shapefile.
- **dist**: Average distance between passes, in meters.
- **max_change_in_heading**: Max change in heading.
- **flag_edges**: Don’t remove edge points, just flag them in the attribute table?.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_remove_off_terrain_objects**

*Remove off terrain objects*

Description

Removes off-terrain objects from a raster digital elevation model (DEM).

Usage

```r
wbt_remove_off_terrain_objects(
  dem,
  output,
  filter = 11,
  slope = 15,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
**wbt_remove_polygon_holes**

**Arguments**

- **dem**: Input raster DEM file.
- **output**: Output raster file.
- **filter**: Filter size (cells).
- **slope**: Slope threshold value.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_remove_polygon_holes**

*Remove polygon holes*

**Description**

Removes holes within the features of a vector polygon file.

**Usage**

```r
wbt_remove_polygon_holes(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input vector polygon file.
- **output**: Output vector polygon file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
wbt_remove_shortStreams

compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

wbt_remove_shortStreams
Remove short streams

Description
Removes short first-order streams from a stream network.

Usage

wbt_remove_shortStreams(
    d8_pntr,
    streams,
    output,
    min_length,
    esri_pntr = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

d8_pntr        Input raster D8 pointer file.
streams        Input raster streams file.
output         Output raster file.
min_length     Minimum tributary length (in map units) used for network pruning.
esri_pntr      D8 pointer uses the ESRI style scheme.
wd             Changes the working directory.
verbose_mode   Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by system() rather than running tool.
**Value**

Returns the tool text outputs.

---

| `wbt_remove_spurs` | **Remove spurs** |

---

**Description**

Removes the spurs (pruning operation) from a Boolean line image; intended to be used on the output of the `LineThinning` tool.

**Usage**

```r
wbt_remove_spurs(
  input,
  output,
  iterations = 10,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `output`: Output raster file.
- `iterations`: Maximum number of iterations.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
Description

This tool resolves topological errors and inconsistencies associated with digitized vector streams.

Usage

```r
wbt_repair_stream_vector_topology(
  input,
  output,
  dist = "",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Name of the input lines vector file.
- **output**: Name of the output lines vector file.
- **dist**: Snap distance, in xy units (metres).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
Resample

Description

Resamples one or more input images into a destination image.

Usage

wbt_resample(
  inputs,
  output,
  cell_size = NULL,
  base = NULL,
  method = "cc",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>inputs</td>
<td>Input raster files.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>cell_size</td>
<td>Optionally specified cell size of output raster. Not used when base raster is specified.</td>
</tr>
<tr>
<td>base</td>
<td>Optionally specified input base raster file. Not used when a cell size is specified.</td>
</tr>
<tr>
<td>method</td>
<td>Resampling method; options include 'nn' (nearest neighbour), 'bilinear', and 'cc' (cubic convolution).</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>

Value

Returns the tool text outputs.
wbt_rescale_value_range

Rescale value range

Description

Performs a min-max contrast stretch on an input greytone image.

Usage

wbt_rescale_value_range(
    input,
    output,
    out_min_val,
    out_max_val,
    clip_min = NULL,
    clip_max = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input Input raster file.
output Output raster file.
out_min_val New minimum value in output image.
out_max_val New maximum value in output image.
clip_min Optional lower tail clip value.
clip_max Optional upper tail clip value.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
wbt_rgb_to_ihs

Description

Converts red, green, and blue (RGB) images into intensity, hue, and saturation (IHS) images.

Usage

```r
wbt_rgb_to_ihs(
  intensity,
  hue,
  saturation,
  red = NULL,
  green = NULL,
  blue = NULL,
  composite = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>intensity</td>
<td>Output intensity raster file.</td>
</tr>
<tr>
<td>hue</td>
<td>Output hue raster file.</td>
</tr>
<tr>
<td>saturation</td>
<td>Output saturation raster file.</td>
</tr>
<tr>
<td>red</td>
<td>Input red band image file. Optionally specified if colour-composite not specified.</td>
</tr>
<tr>
<td>green</td>
<td>Input green band image file. Optionally specified if colour-composite not specified.</td>
</tr>
<tr>
<td>blue</td>
<td>Input blue band image file. Optionally specified if colour-composite not specified.</td>
</tr>
<tr>
<td>composite</td>
<td>Input colour-composite image file. Only used if individual bands are not specified.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>
Value

  Returns the tool text outputs.

---

**Rho8 flow accumulation**

**Description**

This tool calculates Fairfield and Leymarie (1991) flow accumulation.

**Usage**

```r
wbt_rho8_flow_accumulation(
  input,
  output,
  out_type = "specific contributing area",
  log = FALSE,
  clip = FALSE,
  pntr = FALSE,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input DEM or Rho8 pointer file; if a DEM is used, it must be depressionless.
- **output**: Name of the output raster file.
- **out_type**: Output type; one of ‘cells’, ‘specific contributing area’ (default), and ‘catchment area’.
- **log**: Log-transform the output values?.
- **clip**: Optional flag to request clipping the display max by 1 percent.
- **pntr**: Is the input raster a Rho8 flow pointer rather than a DEM?.
- **esri_pntr**: Does the input Rho8 pointer use the ESRI style scheme?.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.
**wbt_rho8_pointer**

**Value**

Returns the tool text outputs.

---

**wbt_rho8_pointer**  
*Rho8 pointer*

**Description**

Calculates a stochastic Rho8 flow pointer raster from an input DEM.

**Usage**

```r
wbt_rho8_pointer(
  dem,  
  output,  
  esri_pntr = FALSE,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dem</td>
<td>Input raster DEM file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>esri_pntr</td>
<td>D8 pointer uses the ESRI style scheme.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.
**wbt_ring_curvature**  
*Ring curvature*

**Description**

This tool calculates ring curvature from an input DEM.

**Usage**

```r
wbt_ring_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **dem** Name of the input raster DEM file.
- **output** Name of the output raster image file.
- **log** Display output values using a log-scale.
- **zfactor** Z conversion factor.
- **wd** Changes the working directory.
- **verbose_mode** Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters** Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only** Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**Description**

Performs a Robert’s cross edge-detection filter on an image.

**Usage**

```r
wbt_roberts_cross_filter(
  input, output,
  clip = 0, wd = NULL,
  verbose_mode = FALSE, compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **clip**: Optional amount to clip the distribution tails by, in percent.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_root_mean_square_error

*Root mean square error*

**Description**

Calculates the RMSE and other accuracy statistics.

**Usage**

```r
wbt_root_mean_square_error(
  input,
  base,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `base`: Input base raster file used for comparison.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

**wbt_rotor**

*Rotor*

**Description**

This tool calculates rotor from an input DEM.
Usage

```r
wbt_rotor(
  dem,
  output,
  log = FALSE,
  zfactor = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

dem
Name of the input raster DEM file.

output
Name of the output raster image file.

log
Display output values using a log-scale.

zfactor
Z conversion factor.

wd
Changes the working directory.

verbose_mode
Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_round**

**Round**

Description

Rounds the values in an input raster to the nearest integer value.

Usage

```r
wbt_round(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>input</strong></td>
<td>Input raster file.</td>
</tr>
<tr>
<td><strong>output</strong></td>
<td>Output raster file.</td>
</tr>
<tr>
<td><strong>wd</strong></td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td><strong>verbose_mode</strong></td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td><strong>compress_rasters</strong></td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td><strong>command_only</strong></td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.

---

**wbt_ruggedness_index**  
**Ruggedness index**

**Description**

Calculates the Riley et al.’s (1999) terrain ruggedness index from an input DEM.

**Usage**

```r
wbt_ruggedness_index(
  dem,  
  output,  
  zfactor = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>dem</strong></td>
<td>Input raster DEM file.</td>
</tr>
<tr>
<td><strong>output</strong></td>
<td>Output raster file.</td>
</tr>
<tr>
<td><strong>zfactor</strong></td>
<td>Optional multiplier for when the vertical and horizontal units are not the same.</td>
</tr>
<tr>
<td><strong>wd</strong></td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td><strong>verbose_mode</strong></td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td><strong>compress_rasters</strong></td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td><strong>command_only</strong></td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>
wbt_run_tool

Value

Returns the tool text outputs.

---------

wbt_run_tool Run a tool in WhiteboxTools by name

Description

Runs a tool and specifies tool arguments. If the prefix "whitebox::" or "wbt_" is in tool_name it is removed to match the definitions in wbt_list_tools()

Usage

wbt_run_tool(tool_name, args, verbose_mode = FALSE, command_only = FALSE)

Arguments

tool_name    The name of the tool to run.
args         Tool arguments.
verbose_mode Verbosity. Without this flag, tool outputs will not be printed.
command_only Return command that would be run with system()? Default: FALSE

Value

Returns the (character) output of the tool.

See Also

wbt_list_tools

Examples

```r
## Not run:
tool_name <- "breach_depressions"
dem <- system.file("extdata", "DEM.tif", package="whitebox")
output <- "./output.tif"
arg1 <- paste0("--dem=", dem)
arg2 <- paste0("--output=", output)
args <- paste(arg1, arg2)
wbt_run_tool(tool_name, args)

## End(Not run)
```
wbt_scharr_filter

Scharr filter

Description

Performs a Scharr edge-detection filter on an image.

Usage

wbt_scharr_filter(
  input,
  output,
  clip = 0,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input Input raster file.
output Output raster file.
clip Optional amount to clip the distribution tails by, in percent.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
Description

Calculates the sediment transport index.

Usage

```r
wbt_sediment_transport_index(
  sca,
  slope,
  output,
  sca_exponent = 0.4,
  slope_exponent = 1.3,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **sca**: Input raster specific contributing area (SCA) file.
- **slope**: Input raster slope file.
- **output**: Output raster file.
- **sca_exponent**: SCA exponent value.
- **slope_exponent**: Slope exponent value.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_select_tiles_by_polygon

Select tiles by polygon

Description

Copies LiDAR tiles overlapping with a polygon into an output directory.

Usage

wbt_select_tiles_by_polygon(
  indir,
  outdir,
  polygons,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

indir  Input LAS file source directory.
outdir Output directory into which LAS files within the polygon are copied.
polygons Input vector polygons file.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_set_nodata_value  Set nodata value

Description

Assign a specified value in an input image to the NoData value.

Usage

```r
wbt_set_nodata_value(
  input,
  output,
  back_value = 0,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `output`: Output raster file.
- `back_value`: Background value to set to nodata.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_shadow_animation  Shadow animation

Description

This tool creates an animated GIF of shadows based on an input DEM.

Usage

```
wbt_shadow_animation(
  input,  # Name of the input digital surface model (DSM) raster file.
  output,  # Name of the output HTML file (*.html).
  palette = "atlas",  # DSM image palette; options are 'atlas', 'high_relief', 'arid', 'soft', 'muted',
  max_dist = "",  # 'light_quant', 'purple', 'viridis', 'gn_yl', 'pi_y_g', 'bl_yl_rd', 'deep', and 'none'.
  date = "21/06/2021",  # Optional maximum search distance, in xy units. Minimum value is 5 x cell size.
  interval = 15,  # Date in format DD/MM/YYYY.
  location = "43.5448/-80.2482/-4",  # Time interval, in minutes (1-60).
  height = 600,  # Location, defined as Lat/Long/UTC-offset (e.g. 43.5448/-80.2482/-4).
  delay = 250,  # Image height, in pixels.
  label = "",  # GIF time delay in milliseconds.
  wd = NULL,  # Label text (leave blank for none).
  verbose_mode = FALSE,  # Changes the working directory.
  compress_rasters = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output mes-
  command_only = FALSE)  # sages.
)
```

Arguments

- **input**: Name of the input digital surface model (DSM) raster file.
- **output**: Name of the output HTML file (*.html).
- **palette**: DSM image palette; options are 'atlas', 'high_relief', 'arid', 'soft', 'muted',
  'light_quant', 'purple', 'viridis', 'gn_yl', 'pi_y_g', 'bl_yl_rd', 'deep', and 'none'.
- **max_dist**: Optional maximum search distance, in xy units. Minimum value is 5 x cell size.
- **date**: Date in format DD/MM/YYYY.
- **interval**: Time interval, in minutes (1-60).
- **location**: Location, defined as Lat/Long/UTC-offset (e.g. 43.5448/-80.2482/-4).
- **height**: Image height, in pixels.
- **delay**: GIF time delay in milliseconds.
- **label**: Label text (leave blank for none).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.
**Value**

Returns the tool text outputs.

| **wbt_shadow_image** | **Shadow image** |

---

**Description**

This tool creates a raster of shadow areas based on an input DEM.

**Usage**

```r
wbt_shadow_image(
  input,
  output,
  palette = "soft",
  max_dist = ",",
  date = "21/06/2021",
  time = "1300",
  location = "43.5448/-80.2482/-4",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Name of the input digital surface model (DSM) raster file.</td>
</tr>
<tr>
<td>output</td>
<td>Name of the output raster file.</td>
</tr>
<tr>
<td>palette</td>
<td>DSM image palette; options are 'atlas', 'high_relief', 'arid', 'soft', 'muted', 'light_quant', 'purple', 'viridi', 'gn_yl', 'pi_y_g', 'bl_yl_rd', 'deep', and 'none'.</td>
</tr>
<tr>
<td>max_dist</td>
<td>Optional maximum search distance, in xy units. Minimum value is 5 x cell size.</td>
</tr>
<tr>
<td>date</td>
<td>Date in format DD/MM/YYYY.</td>
</tr>
<tr>
<td>time</td>
<td>Time in format HH:MM, e.g. 03:15AM or 14:30.</td>
</tr>
<tr>
<td>location</td>
<td>Location, defined as Lat/Long/UTC-offset (e.g. 43.5448/-80.2482/-4).</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>
Value

Returns the tool text outputs.

---

**wbt_shape_complexity_index**

*Shape complexity index*

---

**Description**

Calculates overall polygon shape complexity or irregularity.

**Usage**

```r
wbt_shape_complexity_index(
  input,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input vector polygon file.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.
### wbt_shape_complexity_index_raster

*Shape complexity index raster*

**Description**

Calculates the complexity of raster polygons or classes.

**Usage**

```r
wbt_shape_complexity_index_raster(
  input,  
  output, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input raster file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.

---

### wbt_shape_index

*Shape index*

**Description**

This tool calculates the shape index from an input DEM.
Usage

\[
\text{wbt\_shape\_index(}
\begin{align*}
dem, \\
\text{output,} \\
z\text{factor} = 1, \\
w\text{d} = \text{NULL}, \\
\text{verbose\_mode} = \text{FALSE,} \\
\text{compress\_rasters} = \text{FALSE}, \\
\text{command\_only} = \text{FALSE}
\end{align*}
\)

Arguments

dem \quad \text{Name of the input raster DEM file.}
output \quad \text{Name of the output raster image file.}
z\text{factor} \quad \text{Z conversion factor.}
w\text{d} \quad \text{Changes the working directory.}
verbose\_mode \quad \text{Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.}
compress\_rasters \quad \text{Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.}
command\_only \quad \text{Return command that would be executed by system() rather than running tool.}

Value

Returns the tool text outputs.

---

**wbt\_shreve\_stream\_magnitude**

*Shreve stream magnitude*

---

Description

Assigns the Shreve stream magnitude to each link in a stream network.

Usage

\[
\text{wbt\_shreve\_stream\_magnitude(}
\begin{align*}
d8\_pntr, \\
\text{streams,} \\
\text{output,} \\
es\text{ri\_pntr} = \text{FALSE,} \\
\text{zero\_background} = \text{FALSE,} \\
w\text{d} = \text{NULL},
\end{align*}
\)
Arguments

d8_pntr       Input raster D8 pointer file.
streams       Input raster streams file.
output        Output raster file.
esri_pntr     D8 pointer uses the ESRI style scheme.
zero_background Flag indicating whether a background value of zero should be used.
wd             Changes the working directory.
verbose_mode   Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only   Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Sigmoidal contrast stretch

Usage

wbt_sigmoidal_contrast_stretch(
    input,    
    output,   
    cutoff = 0, 
    gain = 1,  
    num_tones = 256,  
    wd = NULL, 
    verbose_mode = FALSE, 
    compress_rasters = FALSE, 
    command_only = FALSE 
)
Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **cutoff**: Cutoff value between 0.0 and 0.95.
- **gain**: Gain value.
- **num_tones**: Number of tones in the output image.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_sin**

*Sin*

---

Description

Returns the sine (sin) of each values in a raster.

Usage

```r
wbt_sin(   
  input,   
  output,   
  wd = NULL,   
  verbose_mode = FALSE,   
  compress_rasters = FALSE,   
  command_only = FALSE
)
```

Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
**wbt_single_part_to_multi_part**

**Description**
Converts a vector file containing multi-part features into a vector containing only single-part features.

**Usage**
```
wbt_single_part_to_multi_part(
    input,  
    output, 
    field = NULL, 
    wd = NULL, 
    verbose_mode = FALSE, 
    compress_rasters = FALSE, 
    command_only = FALSE
)
```

**Arguments**
- `input` Input vector line or polygon file.
- `output` Output vector line or polygon file.
- `field` Grouping ID field name in attribute table.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**
Returns the tool text outputs.
### wbt_sinh

**Description**

Returns the hyperbolic sine (sinh) of each values in a raster.

**Usage**

```r
wbt_sinh(
  input,  # Input raster file.
  output,  # Output raster file.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE  # Return command that would be executed by system() rather than running tool.
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

### wbt_sink

**Description**

Identifies the depressions in a DEM, giving each feature a unique identifier.
Usage

```r
wbt_sink(
  input,
  output,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `output`: Output raster file.
- `zero_background`: Flag indicating whether a background value of zero should be used.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_slope**

**Slope**

Description

Calculates a slope raster from an input DEM.

Usage

```r
wbt_slope(
  dem,
  output,
  zfactor = NULL,
  units = "degrees",
  wd = NULL,
  verbose_mode = FALSE,
)```
Arguments

dem Input raster DEM file.
output Output raster file.
zfactor Optional multiplier for when the vertical and horizontal units are not the same.
units Units of output raster; options include 'degrees', 'radians', 'percent'.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

This tool creates a slope-aspect relation plot from an input DEM.

Usage

```r
wbt_slope_vs_aspect_plot(
  input, output,
  bin_size = 2, min_slope = 0.1, zfactor = 1, wd = NULL,
  verbose_mode = FALSE, compress_rasters = FALSE, command_only = FALSE
)
```
**Arguments**

- **input**: Name of the input raster image file.
- **output**: Name of the output report file (*.html).
- **bin_size**: Aspect bin size, in degrees.
- **min_slope**: Minimum slope, in degrees.
- **zfactor**: Z conversion factor.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**Description**

Creates a slope vs. elevation plot for one or more DEMs.

**Usage**

```r
wbt_slope_vs_elevation_plot(
  inputs,
  output,
  watershed = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **inputs**: Input DEM files.
- **output**: Output HTML file (default name will be based on input file if unspecified).
- **watershed**: Input watershed files (optional).
wd
verbose_mode
compress_rasters
command_only

Changes the working directory.
Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

Description
Smoothes a vector coverage of either a POLYLINE or POLYGON base ShapeType.

Usage
wbt_smooth_vectors(
  input,
  output,
  filter = 3,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments
input Input vector POLYLINE or POLYGON file.
output Output vector file.
filter The filter size, any odd integer greater than or equal to 3; default is 3.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.
This tool can smooth the residual roughness due to vegetation cover in LiDAR DEMs.

Usage

```r
wbt_smooth_vegetation_residual(
    input, output,
    max_scale = 30,
    dev_threshold = 1,
    scale_threshold = 5,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

Arguments

- **input**: Name of the input digital elevation model (DEM) raster file.
- **output**: Name of the output raster file.
- **max_scale**: Maximum search neighbourhood radius in grid cells.
- **dev_threshold**: DEVmax Threshold.
- **scale_threshold**: DEVmax scale threshold.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**wbt_snap_pour_points**  

_Snap pour points_

### Description

Moves outlet points used to specify points of interest in a watershedding operation to the cell with the highest flow accumulation in its neighbourhood.

### Usage

```r
wbt_snap_pour_points(
  pour_pts,
  flow_accum,
  output,
  snap_dist,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

- **pour_pts**: Input vector pour points (outlet) file.
- **flow_accum**: Input raster D8 flow accumulation file.
- **output**: Output vector file.
- **snap_dist**: Maximum snap distance in map units.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

### Value

Returns the tool text outputs.
wbt_sobel_filter

Description

Performs a Sobel edge-detection filter on an image.

Usage

```r
wbt_sobel_filter(
  input,
  output,
  variant = "3x3",
  clip = 0,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input raster file.
- **output**: Output raster file.
- **variant**: Optional variant value. Options include 3x3 and 5x5 (default is 3x3).
- **clip**: Optional amount to clip the distribution tails by, in percent (default is 0.0).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_spherical_std_dev_of_normals

*Spherical std dev of normals*

**Description**

Calculates the spherical standard deviation of surface normals for a DEM.

**Usage**

```r
wbt_spherical_std_dev_of_normals(
  dem,
  output,
  filter = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `output`: Output raster file.
- `filter`: Size of the filter kernel.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_split_colour_composite

Split colour composite

Description

This tool splits an RGB colour composite image into separate multispectral images.

Usage

```r
wbt_split_colour_composite(
  input,
  red = NULL,
  green = NULL,
  blue = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input colour composite image file.
- **red**: Output red band file.
- **green**: Output green band file.
- **blue**: Output blue band file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**wbt_split_vector_lines**

*Split vector lines*

**Description**

This tool can be used to split a vector line coverage into even-lengthed segments.

**Usage**

```r
wbt_split_vector_lines(
  input,  # Name of the input lines shapefile.
  output,  # Name of the output lines shapefile.
  length = NULL,  # Maximum segment length (m).
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE  # Return command that would be executed by system() rather than running tool.
)
```

**Arguments**

- **input**: Name of the input lines shapefile.
- **output**: Name of the output lines shapefile.
- **length**: Maximum segment length (m).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.
Description

Splits the lines or polygons in one layer using the lines in another layer.

Usage

```r
wbt_split_with_lines(
  input,
  split,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input vector line or polygon file.
- **split**: Input vector polyline file.
- **output**: Output vector file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_square  

**Description**

Squares the values in a raster.

**Usage**

```r
wbt_square(
  input,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_square_root  

**Description**

Returns the square root of the values in a raster.
Usage

wbt_square_root(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input       Input raster file.
output      Output raster file.
wd          Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_standard_deviation_contrast_stretch

Standard deviation contrast stretch

Description

Performs a standard-deviation contrast stretch on input images.

Usage

wbt_standard_deviation_contrast_stretch(
    input,
    output,
    stdev = 2,
    num_tones = 256,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
Arguments

input  Input raster file.
output  Output raster file.
stdev  Standard deviation clip value.
num_tones  Number of tones in the output image.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_standard_deviation_filter**

*Standard deviation filter*

Description

Assigns each cell in the output grid the standard deviation of values in a moving window centred on each grid cell in the input raster.

Usage

```r
wbt_standard_deviation_filter(
  input,  # Input raster file.
  output,  # Output raster file.
  filterx = 11,  # Filter size in the x-direction.
  filtery = 11,  # Filter size in the y-direction.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE  # Return command that would be executed by system() rather than running tool.
)
```
Arguments

input    Input raster file.
output   Output raster file.
filterx  Size of the filter kernel in the x-direction.
filtery  Size of the filter kernel in the y-direction.
wd       Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Calculates the standard deviation of slope from an input DEM.

Usage

wbt_standard_deviation_of_slope(
    input,
    output,
    zfactor = NULL,
    filterx = 11,
    filtery = 11,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
Arguments

- **input**: Input raster DEM file.
- **output**: Output raster DEM file.
- **zfactor**: Optional multiplier for when the vertical and horizontal units are not the same.
- **filterx**: Size of the filter kernel in the x-direction.
- **filtery**: Size of the filter kernel in the y-direction.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**Description**

Performs a stochastic analysis of depressions within a DEM.

**Usage**

```r
wbt_stochastic_depression_analysis(
  dem,
  output,
  rmse,
  range,
  iterations = 100,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
Arguments

- **dem**: Input raster DEM file.
- **output**: Output file.
- **rmse**: The DEM’s root-mean-square-error (RMSE), in z units. This determines error magnitude.
- **range**: The error field’s correlation length, in xy-units.
- **iterations**: The number of iterations.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_strahler_order_basins**

*Strahler order basins*

Description

Identifies Strahler-order basins from an input stream network.

Usage

```r
wbt_strahler_order_basins(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```
wbt_strahler_stream_order

**Arguments**

- `streams` : Input raster streams file.
- `output` : Output raster file.
- `esri_pntr` : D8 pointer uses the ESRI style scheme.
- `wd` : Changes the working directory.
- `verbose_mode` : Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` : Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` : Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

**Description**

Assigns the Strahler stream order to each link in a stream network.

**Usage**

```r
wbt_strahler_stream_order(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `streams` : Input raster streams file.
- `output` : Output raster file.
**Description**

Identifies the exterior/interior links and nodes in a stream network.

**Usage**

```r
wbt_stream_link_class(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `d8_pntr` Input raster D8 pointer file.
- `streams` Input raster streams file.
- `output` Output raster file.
- `esri_pntr` D8 pointer uses the ESRI style scheme.
- `zero_background` Flag indicating whether a background value of zero should be used.
- `wd` Changes the working directory.

**Value**

Returns the tool text outputs.
**wbt_stream_link_identifier**

**Stream link identifier**

**Description**

Assigns a unique identifier to each link in a stream network.

**Usage**

```r
wbt_stream_link_identifier(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **d8_pntr**  
  Input raster D8 pointer file.

- **streams**  
  Input raster streams file.

- **output**  
  Output raster file.

- **esri_pntr**  
  D8 pointer uses the ESRI style scheme.

- **zero_background**  
  Flag indicating whether a background value of zero should be used.

- **wd**  
  Changes the working directory.

- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

- **compress_rasters**  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

- **command_only**  
  Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.


**wbt_stream_link_length**

`compress_rasters`  
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

`command_only`  
Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

**Description**

Estimates the length of each link (or tributary) in a stream network.

**Usage**

```r
wbt_stream_link_length(
  d8_pntr,  
  linkid,  
  output,  
  esri_pntr = FALSE,  
  zero_background = FALSE,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)
```

**Arguments**

- `d8_pntr`  
  Input raster D8 pointer file.
- `linkid`  
  Input raster streams link ID (or tributary ID) file.
- `output`  
  Output raster file.
- `esri_pntr`  
  D8 pointer uses the ESRI style scheme.
- `zero_background`  
  Flag indicating whether a background value of zero should be used.
- `wd`  
  Changes the working directory.
- `verbose_mode`  
  Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`  
  Return command that would be executed by `system()` rather than running tool.
wbt_stream_link_slope

Value

Returns the tool text outputs.

wbt_stream_link_slope  Stream link slope

Description

Estimates the average slope of each link (or tributary) in a stream network.

Usage

wbt_stream_link_slope(
  d8_pntr,
  linkid,
  dem,
  output,
  esri_pntr = FALSE,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

d8_pntr  Input raster D8 pointer file.
linkid   Input raster streams link ID (or tributary ID) file.
dem      Input raster DEM file.
output   Output raster file.
esri_pntr D8 pointer uses the ESRI style scheme.
zero_background Flag indicating whether a background value of zero should be used.
wd       Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
**wbt_stream_power_index**

*Stream power index*

**Description**

Calculates the relative stream power index.

**Usage**

```r
wbt_stream_power_index(
  sca,  
slope,  
output,  
exponent = 1,  
wd = NULL,  
verbose_mode = FALSE,  
compress_rasters = FALSE,  
command_only = FALSE
)
```

**Arguments**

- `sca` Input raster specific contributing area (SCA) file.
- `slope` Input raster slope file.
- `output` Output raster file.
- `exponent` SCA exponent value.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_stream_slope_continuous

Stream slope continuous

Description
Estimates the slope of each grid cell in a stream network.

Usage

wbt_stream_slope_continuous(
  d8_pntr,
  streams,
  dem,
  output,
  esri_pntr = FALSE,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

streams: Input raster streams file.
output: Output raster file.
esri_pntr: D8 pointer uses the ESRI style scheme.
zero_background: Flag indicating whether a background value of zero should be used.
wd: Changes the working directory.
verbose_mode: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only: Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.
wbt_subbasins

wbt_subbasins  Subbasins

Description

Identifies the catchments, or sub-basin, draining to each link in a stream network.

Usage

wbt_subbasins(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

d8_pntr       Input D8 pointer raster file.
streams       Input raster streams file.
output        Output raster file.
esri_pntr     D8 pointer uses the ESRI style scheme.
wd             Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters       Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
Description

Performs a differencing operation on two rasters or a raster and a constant value.

Usage

```r
wbt_subtract(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `input1`: Input raster file or constant value.
- `input2`: Input raster file or constant value.
- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**wbt_sum_overlay**  

description

Calculates the sum for each grid cell from a group of raster images.

**Usage**

```r
wbt_sum_overlay(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>inputs</td>
<td>Input raster files.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by <code>system()</code> rather than running tool.</td>
</tr>
</tbody>
</table>

**Value**

Returns the tool text outputs.

---

**wbt_surface_area_ratio**  

description

Calculates a the surface area ratio of each grid cell in an input DEM.
Usage

```r
wbt_surface_area_ratio(
  dem,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

output: Output raster file.
wd: Changes the working directory.
verbose_mode: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
compress_rasters: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_svm_classification**

*Svm classification*

Description

Performs an SVM binary classification using training site polygons/points and multiple input images.

Usage

```r
wbt_svm_classification(
  inputs,
  training,
  field,
  scaling = "Normalize",
  output = NULL,
  c = 200,
  gamma = 50,
)```
tolerance = 0.1,
test_proportion = 0.2,
wd = NULL,
verbose_mode = FALSE,
compress_rasters = FALSE,
command_only = FALSE
)

Arguments

inputs Names of the input predictor rasters.
training Name of the input training site polygons/points Shapefile.
field Name of the attribute containing class data.
scaling Scaling method for predictors. Options include 'None', 'Normalize', and 'Standardize'.
output Name of the output raster file.
c c-value, the regularization parameter.
gamma Gamma parameter used in setting the RBF (Gaussian) kernel function.
tolerance The tolerance parameter used in determining the stopping condition.
test_proportion The proportion of the dataset to include in the test split; default is 0.2.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_svm_regression  Svm regression

Description

Performs a supervised SVM regression analysis using training site points and predictor rasters.
**Usage**

```r
wbt_svm_regression(
    inputs,
    training,
    field,
    scaling = "Normalize",
    output = NULL,
    c = 50,
    eps = 10,
    gamma = 0.5,
    test_proportion = 0.2,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

- **inputs**: Names of the input predictor rasters.
- **training**: Name of the input training site points Shapefile.
- **field**: Name of the attribute containing class data.
- **scaling**: Scaling method for predictors. Options include 'None', 'Normalize', and 'Standardize'.
- **output**: Name of the output raster file.
- **c**: c-value, the regularization parameter.
- **eps**: Epsilon in the epsilon-SVR model.
- **gamma**: Gamma parameter used in setting the RBF (Gaussian) kernel function.
- **test_proportion**: The proportion of the dataset to include in the test split; default is 0.2.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**Description**

Outputs the features that occur in one of the two vector inputs but not both, i.e. no overlapping features.

**Usage**

```r
wbt_symmetrical_difference(
  input,
  overlay,
  output,
  snap = 0,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input vector file.
- **overlay**: Input overlay vector file.
- **output**: Output vector file.
- **snap**: Snap tolerance.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
### wbt_tan

**Tan**

**Description**

Returns the tangent (tan) of each values in a raster.

**Usage**

```r
wbt_tan(
  input,  
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

### wbt_tangential_curvature

*Tangential curvature*

**Description**

Calculates a tangential curvature raster from an input DEM.
wbt_tanh

Usage

wbt_tangential_curvature(
    dem,
    output,
    log = FALSE,
    zfactor = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

dem Input raster DEM file.
output Output raster file.
log Display output values using a log-scale.
zfactor Optional multiplier for when the vertical and horizontal units are not the same.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_tanh  Tanh

Description

Returns the hyperbolic tangent (tanh) of each values in a raster.

Usage

wbt_tanh(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
wbt_thicken_raster_line

Arguments

input  Input raster file.
output Output raster file.
wd     Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_thicken_raster_line

Thicken raster line

Description

Thickens single-cell wide lines within a raster image.

Usage

wbt_thicken_raster_line(
   input,
   output,
   wd = NULL,
   verbose_mode = FALSE,
   compress_rasters = FALSE,
   command_only = FALSE
)

Arguments

input  Input raster file.
output Output raster file.
wd     Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.
Value

Returns the tool text outputs.

\[wbt\_time\_in\_daylight\]

**Description**

Calculates the proportion of time a location is not within an area of shadow.

**Usage**

```r
wbt_time_in_daylight(
    dem,  
    output,  
    lat,  
    long,  
    az_fraction = 10,  
    max_dist = 100,  
    utc_offset = "0000",  
    start_day = 1,  
    end_day = 365,  
    start_time = "000000",  
    end_time = "235959",  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE
)
```

**Arguments**

dem Input raster DEM file.
output Output raster file.
lat Centre point latitude.
long Centre point longitude.
az_fraction Azimuth fraction in degrees.
max_dist Optional maximum search distance. Minimum value is 5 x cell size.
utc_offset UTC time offset, in hours (e.g. -04:00, +06:00).
start_day Start day of the year (1-365).
end_day End day of the year (1-365).
start_time Starting hour to track shadows (e.g. 5, 5:00, 05:00:00). Assumes 24-hour time: HH:MM:SS. 'sunrise' is also a valid time.
Description

Creates a raster grid based on a triangular irregular network (TIN) fitted to vector points.

Usage

```r
wbt_tin_gridding(
    input, output,
    field = NULL,
    use_z = FALSE,
    resolution = NULL,
    base = NULL,
    max_triangle_edge_length = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

Arguments

- **input**: Input vector points file.
- **output**: Output raster file.
- **field**: Input field name in attribute table.
- **use_z**: Use the 'z' dimension of the Shapefile's geometry instead of an attribute field?.
- **resolution**: Output raster's grid resolution.
- **base**: Optionally specified input base raster file. Not used when a cell size is specified.
max_triangle_edge_length
Optional maximum triangle edge length; triangles larger than this size will not be gridded.

wd
Changes the working directory.

verbose_mode
Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only
Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

---

wbt_toolbox
The toolbox for a specific tool in WhiteboxTools

Description
Retrieve the toolbox for a specific tool.

Usage
wbt_toolbox(tool_name = NULL)

Arguments
tool_name The name of the tool.

Details
Leaving tool_name as default NULL returns results for all tools, but does not work on Windows.

Value
Returns the toolbox for a specific tool.

Examples
## Not run:
wbt_toolbox("breach_depressions")

## End(Not run)
wbt_tool_help

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieves the help description for a specific tool.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>wbt_tool_help(tool_name = NULL)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Arguments</th>
</tr>
</thead>
<tbody>
<tr>
<td>tool_name</td>
</tr>
<tr>
<td>The name of the tool.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leaving tool_name as default NULL returns results for all tools, but does not work on Windows.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Returns the help description for a specific tool.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>## Not run:</td>
</tr>
<tr>
<td>wbt_tool_help(&quot;lidar_info&quot;)</td>
</tr>
<tr>
<td>## End(Not run)</td>
</tr>
</tbody>
</table>

wbt_tool_parameters

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieves the tool parameter descriptions for a specific tool.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>wbt_tool_parameters(tool_name, quiet = FALSE)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Arguments</th>
</tr>
</thead>
<tbody>
<tr>
<td>tool_name</td>
</tr>
<tr>
<td>The name of the tool.</td>
</tr>
<tr>
<td>quiet</td>
</tr>
<tr>
<td>Prevent tool output being printed. Default: FALSE</td>
</tr>
</tbody>
</table>
### wbt_tophat_transform

**Details**

quiet argument can be set to TRUE to allow for "quiet" internal use within other functions.

**Value**

Returns the tool parameter descriptions for a specific tool.

**Examples**

```r
## Not run:
wbt_tool_parameters("lidar_info")
## End(Not run)
```

---

**wbt_tophat_transform**  
**Top hat transform**

**Description**

Performs either a white or black top-hat transform on an input image.

**Usage**

```r
wbt_tophat_transform(
    input,  
    output,  
    filterx = 11,  
    filtery = 11,  
    variant = "white",  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **filterx**: Size of the filter kernel in the x-direction.
- **filtery**: Size of the filter kernel in the y-direction.
- **variant**: Optional variant value. Options include 'white' and 'black'.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only
Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

wbt_topographic_position_animation

Description
This tool creates an animated GIF of multi-scale local topographic position (elevation deviation).

Usage
wbt_topographic_position_animation(
  input,
  output,
  palette = "bl_yl_rd",
  min_scale = 1,
  num_steps = 100,
  step_nonlinearity = 1.5,
  height = 600,
  delay = 250,
  label = "",
  dev_max = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments
input Name of the input digital elevation model (DEM) raster file.
output Name of the output HTML file (*.html).
palette Image palette; options are 'bl_yl_rd', 'bl_w_rd', 'purple', 'gn_yl', 'pi_y_g', and 'viridis'.
min_scale Minimum search neighbourhood radius in grid cells.
um_steps Number of steps.
step_nonlinearity Step nonlinearity factor (1.0-2.0 is typical).
wbt_topological_stream_order

**Description**

Assigns each link in a stream network its topological order.

**Usage**

```r
wbt_topological_stream_order(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **d8_pntr**: Input raster D8 pointer file.
- **streams**: Input raster streams file.
- **output**: Output raster file.

**Value**

Returns the tool text outputs.
esri_pntr  D8 pointer uses the ESRI style scheme.
zero_background  Flag indicating whether a background value of zero should be used.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.

---

**wbt_total_curvature**  *Total curvature*

**Description**
Calculates a total curvature raster from an input DEM.

**Usage**

```r
wbt_total_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **dem**  Input raster DEM file.
- **output**  Output raster file.
- **log**  Display output values using a log-scale.
- **zfactor**  Optional multiplier for when the vertical and horizontal units are not the same.
- **wd**  Changes the working directory.
- **verbose_mode**  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
**wbt_total_filter**

wbt_total_filter

**compress_rasters**
Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

**command_only**
Return command that would be executed by `system()` rather than running tool.

**Value**
Returns the tool text outputs.

---

**wbt_total_filter**

**Total filter**

---

**Description**
Performs a total filter on an input image.

**Usage**

```r
wbt_total_filter(
  input,
  output,
  filterx = 11,
  filtery = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **filterx**: Size of the filter kernel in the x-direction.
- **filtery**: Size of the filter kernel in the y-direction.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**
Returns the tool text outputs.
## WBT_to_degrees

**To degrees**

**Description**

Converts a raster from radians to degrees.

**Usage**

```r
wbt_to_degrees(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input`  
  Input raster file.
- `output`  
  Output raster file.
- `wd`  
  Changes the working directory.
- `verbose_mode`  
  Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`  
  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

## WBT_to_radians

**To radians**

**Description**

Converts a raster from degrees to radians.
**Usage**

```r
wbt_to_radians(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

- **input**
  - Input raster file.
- **output**
  - Output raster file.
- **wd**
  - Changes the working directory.
- **verbose_mode**
  - Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**
  - Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**
  - Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

**wbt_trace_downslope_flowpaths**

*Trace downslope flowpaths*

---

**Description**

Traces downslope flowpaths from one or more target sites (i.e. seed points).

**Usage**

```r
wbt_trace_downslope_flowpaths(
    seed_pts,
    d8_pntr,
    output,
    esri_pntr = FALSE,
    zero_background = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```
wbt_trend_surface

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>seed_pts</td>
<td>Input vector seed points file.</td>
</tr>
<tr>
<td>d8_pntr</td>
<td>Input D8 pointer raster file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>esri_pntr</td>
<td>D8 pointer uses the ESRI style scheme.</td>
</tr>
<tr>
<td>zero_background</td>
<td>Flag indicating whether a background value of zero should be used.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>

Value

Returns the tool text outputs.

wbt_trend_surface  Trend surface

Description

Estimates the trend surface of an input raster file.

Usage

```r
wbt_trend_surface(
  input,
  output,
  order = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Input raster file.</td>
</tr>
<tr>
<td>output</td>
<td>Output raster file.</td>
</tr>
<tr>
<td>order</td>
<td>Polynomial order (1 to 10).</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
</tbody>
</table>
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

---

**Description**

Estimates a trend surface from vector points.

**Usage**

```r
wbt_trend_surface_vector_points(
  input,
  field,
  output,
  cell_size,
  order = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input**  Input vector Points file.
- **field**  Input field name in attribute table.
- **output**  Output raster file.
- **cell_size**  Optionally specified cell size of output raster. Not used when base raster is specified.
- **order**  Polynomial order (1 to 10).
- **wd**  Changes the working directory.
- **verbose_mode**  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
```
compress_rasters
   Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only
   Return command that would be executed by `system()` rather than running tool.
```

**Value**

Returns the tool text outputs.

---

### `wbt_tributary_identifier`

*Tributary identifier*

**Description**

Assigns a unique identifier to each tributary in a stream network.

**Usage**

```
wbt_tributary_identifier(
   d8_pntr,
   streams,
   output,
   esri_pntr = FALSE,
   zero_background = FALSE,
   wd = NULL,
   verbose_mode = FALSE,
   compress_rasters = FALSE,
   command_only = FALSE
)
```

**Arguments**

- `streams`: Input raster streams file.
- `output`: Output raster file.
- `esri_pntr`: D8 pointer uses the ESRI style scheme.
- `zero_background`: Flag indicating whether a background value of zero should be used.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.
**Value**

Returns the tool text outputs.

---

**wbtt_truncate**  
**Truncate**

**Description**

Truncates the values in a raster to the desired number of decimal places.

**Usage**

```r
wbtt_truncate(
  input,  
  output,  
  num_decimals = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE
)
```

**Arguments**

- **input**: Input raster file.
- **output**: Output raster file.
- **num_decimals**: Number of decimals left after truncation (default is zero).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_turning_bands_simulation

*Turning bands simulation*

**Description**

Creates an image containing random values based on a turning-bands simulation.

**Usage**

```r
wbt_turning_bands_simulation(
  base,
  output,
  range,
  iterations = 1000,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **base**: Input base raster file.
- **output**: Output file.
- **range**: The field’s range, in xy-units, related to the extent of spatial autocorrelation.
- **iterations**: The number of iterations.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
Two sample ks test

Description

Performs a 2-sample K-S test for significant differences on two input rasters.

Usage

```r
wbt_two_sample_ks_test(
  input1, 
  input2,
  output,
  num_samples = NULL, 
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input1**: First input raster file.
- **input2**: Second input raster file.
- **output**: Output HTML file.
- **num_samples**: Number of samples. Leave blank to use whole image.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
Description

Splits vector layers at their overlaps, creating a layer containing all the portions from both input and overlay layers.

Usage

```r
wbt_union(
  input,  
  overlay, 
  output, 
  snap = 0, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)
```

Arguments

- **input**: Input vector file.
- **overlay**: Input overlay vector file.
- **output**: Output vector file.
- **snap**: Snap tolerance.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
Description

Extract whole watersheds for a set of outlet points.

Usage

```r
wbt_unnest_basins(
  d8_pntr,
  pour_pts,
  output,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `pour_pts`: Input vector pour points (outlet) file.
- `output`: Output raster file.
- `esri_pntr`: D8 pointer uses the ESRI style scheme.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
Unsharp masking

Description
An image sharpening technique that enhances edges.

Usage
wbt_unsharp_masking(  
  input,  
  output,  
  sigma = 0.75,  
  amount = 100,  
  threshold = 0,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)

Arguments

input Input raster file.
output Output raster file.
sigma Standard deviation distance in pixels.
amount A percentage and controls the magnitude of each overshoot.
threshold Controls the minimal brightness change that will be sharpened.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.
Description

This tool calculates the unsphericity curvature from an input DEM.

Usage

```r
wbt_unsphericity(
  dem,
  output,
  log = FALSE,
  zfactor = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

dem       Name of the input raster DEM file.
output    Name of the output raster image file.
log       Display output values using a log-scale.
zfactor  Z conversion factor.
wd        Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_update_nodata_cells

Update nodata cells

Description

Replaces the NoData values in an input raster with the corresponding values contained in a second update layer.

Usage

wbt_update_nodata_cells(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input1  Input raster file 1.
input2  Input raster file 2; update layer.
output  Output raster file.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_upslope_depression_storage

*Upslope depression storage*

**Description**

Estimates the average upslope depression storage depth.

**Usage**

```r
wbt_upslope_depression_storage(
  dem,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_user_defined_weights_filter

*User ined weights filter*

**Description**

Performs a user-defined weights filter on an image.
Usage

```r
wbt_user_defined_weights_filter(
  input,
  weights,
  output,
  center = "center",
  normalize = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input raster file.
- **weights**: Input weights file.
- **output**: Output raster file.
- **center**: Kernel center cell; options include 'center', 'upper-left', 'upper-right', 'lower-left', 'lower-right'.
- **normalize**: Normalize kernel weights? This can reduce edge effects and lessen the impact of data gaps (nodata) but is not suited when the kernel weights sum to zero.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

Description

Hex-bins a set of vector points.
Usage

```r
wbt_vector_hex_binning(
  input,
  output,
  width,
  orientation = "horizontal",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `output`: Output vector polygon file.
- `width`: The grid cell width.
- `orientation`: Grid Orientation, 'horizontal' or 'vertical'.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_vector_lines_to_raster**

*Vector lines to raster*

Description

Converts a vector containing polylines into a raster.

Usage

```r
wbt_vector_lines_to_raster(
  input,
  output,
  field = "FID",
  nodata = TRUE,
)```
wbt_vector_points_to_raster

Vector points to raster

Description

Converts a vector containing points into a raster.

Usage

wbt_vector_points_to_raster(input, output, field = "FID", assign = "last", nodata = TRUE,
Arguments

input: Input vector Points file.

output: Output raster file.

field: Input field name in attribute table.

assign: Assignment operation, where multiple points are in the same grid cell; options include 'first', 'last' (default), 'min', 'max', 'sum'.

nodata: Background value to set to NoData. Without this flag, it will be set to 0.0.

cell_size: Optionally specified cell size of output raster. Not used when base raster is specified.

base: Optionally specified input base raster file. Not used when a cell size is specified.

wd: Changes the working directory.

verbose_mode: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

command_only: Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

Description

Converts a vector containing polygons into a raster.
Usage

wbt_vector_polygons_to_raster(
    input, output,
    field = "FID",
    nodata = TRUE,
    cell_size = NULL,
    base = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input Input vector polygons file.
output Output raster file.
field Input field name in attribute table.
nodata Background value to set to NoData. Without this flag, it will be set to 0.0.
cell_size Optionally specified cell size of output raster. Not used when base raster is specified.
base Optionally specified input base raster file. Not used when a cell size is specified.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_vector_stream_network_analysis
Vector stream analysis

Description

This tool performs common stream network analysis operations on an input vector stream file.
Usage

wbt_vector_stream_network_analysis(
    streams,
    dem,
    output,
    cutting_height = 10,
    snap = 0.1,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>streams</td>
<td>Name of the input streams vector file.</td>
</tr>
<tr>
<td>dem</td>
<td>Name of the input DEM raster file.</td>
</tr>
<tr>
<td>output</td>
<td>Name of the output lines shapefile.</td>
</tr>
<tr>
<td>cutting_height</td>
<td>Maximum ridge-cutting height (z units).</td>
</tr>
<tr>
<td>snap</td>
<td>Snap distance, in xy units (metres).</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>

Value

Returns the tool text outputs.

wbt_version

Version information for WhiteboxTools

Description

Version information for WhiteboxTools

Usage

wbt_version()

Value

Returns the version information for WhiteboxTools as an R character vector.
wbt_vertical_excess_curvature

Vertical excess curvature

Description

This tool calculates vertical excess curvature from an input DEM.

Usage

wbt_vertical_excess_curvature(
  dem,                    
  output,                 
  log = FALSE,            
  zfactor = 1,            
  wd = NULL,              
  verbose_mode = FALSE,   
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

dem Name of the input raster DEM file.
output Name of the output raster image file.
log Display output values using a log-scale.
zfactor Z conversion factor.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
Description

Identifies the viewshed for a point or set of points.

Usage

```r
wbt_viewshed(
  dem,
  stations,
  output,
  height = 2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

dem Input raster DEM file.
stations Input viewing station vector file.
output Output raster file.
height Viewing station height, in z units.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
**wbt_view_code**  
*Source code for a specific tool in WhiteboxTools*

### Description

Opens a web browser to view the source code for a specific tool on the project's source code repository.

### Usage

```r
wbt_view_code(tool_name, viewer = FALSE)
```

### Arguments

- **tool_name**: Name of the tool.
- **viewer**: Show source code in browser? default: TRUE

### Value

Returns a GitHub URL to view the source code of the tool.

### Examples

```r
## Not run:
wbt_view_code("breach_depressions")
## End(Not run)
```

**wbt_visibility_index**  
*Visibility index*

### Description

Estimates the relative visibility of sites in a DEM.

### Usage

```r
wbt_visibility_index(
  dem, 
  output, 
  height = 2, 
  res_factor = 2, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE
)
```
wbt_voronoi_diagram

Arguments

dem Input raster DEM file.
output Output raster file.
height Viewing station height, in z units.
res_factor The resolution factor determines the density of measured viewsheds.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_voronoi_diagram Voronoi diagram

Description

Creates a vector Voronoi diagram for a set of vector points.

Usage

wbt_voronoi_diagram(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

input Input vector points file.
output Output vector polygon file.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.
Value

Returns the tool text outputs.

---

wbt_watershed  

**Watershed**

**Description**

Identifies the watershed, or drainage basin, draining to a set of target cells.

**Usage**

```r
wbt_watershed(
  d8_pntr,
  pour_pts,
  output,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **d8_pntr**  
  Input D8 pointer raster file.

- **pour_pts**  
  Input pour points (outlet) file.

- **output**  
  Output raster file.

- **esri_pntr**  
  D8 pointer uses the ESRI style scheme.

- **wd**  
  Changes the working directory.

- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

- **compress_rasters**  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.

- **command_only**  
  Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_weighted_overlay  Weighted overlay

Description
Performs a weighted sum on multiple input rasters after converting each image to a common scale. The tool performs a multi-criteria evaluation (MCE).

Usage
```r
wbt_weighted_overlay(
  factors,  # Input factor raster files.
  weights,  # Weight values, contained in quotes and separated by commas or semicolons. Must have the same number as factors.
  output,   # Output raster file.
  cost = NULL,  # Weight values, contained in quotes and separated by commas or semicolons. Must have the same number as factors.
  constraints = NULL,  # Input constraints raster files.
  scale_max = 1,  # Suitability scale maximum value (common values are 1.0, 100.0, and 255.0).
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments
- **factors**: Input factor raster files.
- **weights**: Weight values, contained in quotes and separated by commas or semicolons. Must have the same number as factors.
- **output**: Output raster file.
- **cost**: Weight values, contained in quotes and separated by commas or semicolons. Must have the same number as factors.
- **constraints**: Input constraints raster files.
- **scale_max**: Suitability scale maximum value (common values are 1.0, 100.0, and 255.0).
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by system() rather than running tool.

Value
Returns the tool text outputs.
wbt_weighted_sum  Weighted sum

Description

Performs a weighted-sum overlay on multiple input raster images.

Usage

```r
wbt_weighted_sum(
  inputs,
  weights,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- `inputs`  Input raster files.
- `weights`  Weight values, contained in quotes and separated by commas or semicolons.
- `output`  Output raster file.
- `wd`  Changes the working directory.
- `verbose_mode`  Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- `compress_rasters`  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`  Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_wetness_index  

**Wetness index**

Description

Calculates the topographic wetness index, $\ln(A / \tan(slope))$.

Usage

```r
wbt_wetness_index(
  sca,  # Input raster specific contributing area (SCA) file.
  slope,  # Input raster slope file (in degrees).
  output,  # Output raster file.
  wd = NULL,  # Changes the working directory.
  verbose_mode = FALSE,  # Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
  compress_rasters = FALSE,  # Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
  command_only = FALSE  # Return command that would be executed by `system()` rather than running tool.
)
```

Arguments

- **sca**: Input raster specific contributing area (SCA) file.
- **slope**: Input raster slope file (in degrees).
- **output**: Output raster file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.
wbt_wilcoxon_signed_rank_test

*Wilcoxon signed rank test*

**Description**

Performs a 2-sample K-S test for significant differences on two input rasters.

**Usage**

```r
wbt_wilcoxon_signed_rank_test(
  input1,
  input2,
  output,
  num_samples = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input1**: First input raster file.
- **input2**: Second input raster file.
- **output**: Output HTML file.
- **num_samples**: Number of samples. Leave blank to use whole image.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_write_function_memory_insertion**

*Write function memory insertion*

**Description**

Performs a write function memory insertion for single-band multi-date change detection.

**Usage**

```r
wbt_write_function_memory_insertion(
  input1,
  input2,
  output,
  input3 = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- **input1**: Input raster file associated with the first date.
- **input2**: Input raster file associated with the second date.
- **output**: Output raster file.
- **input3**: Optional input raster file associated with the third date.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**Description**

Performs a logical XOR operator on two Boolean raster images.

**Usage**

```r
wbt_xor(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input1`: Input raster file.
- `input2`: Input raster file.
- `output`: Output raster file.
- `wd`: Changes the working directory.
- `verbose_mode`: Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters`: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only`: Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
**wbt_yield_filter**  
*Yield filter*

**Description**
Filters crop yield values of point data derived from combine harvester yield monitors.

**Usage**
```
wbt_yield_filter(  
    input,  
    yield_field,  
    pass_field,  
    output,  
    width = 6.096,  
    z_score_threshold = 2.5,  
    min_yield = 0,  
    max_yield = 99999.9,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**
- **input**  
  Name of the input points shapefile.
- **yield_field**  
  Name of the attribute containing yield data.
- **pass_field**  
  Name of the attribute containing pass line ID.
- **output**  
  Name of the output points shapefile.
- **width**  
  Pass swath width (m).
- **z_score_threshold**  
  Z-score threshold value (default=2.5).
- **min_yield**  
  Minimum yield value in output.
- **max_yield**  
  Maximum yield value in output.
- **wd**  
  Changes the working directory.
- **verbose_mode**  
  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- **compress_rasters**  
  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**  
  Return command that would be executed by system() rather than running tool.

**Value**
Returns the tool text outputs.
**Description**

This tool can be used to create a segmented-vector polygon yield map from a set of harvester points.

**Usage**

```r
wbt_yield_map(
  input,
  pass_field_name,
  output,
  width = 6.096,
  max_change_in_heading = 25,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

- `input` Name of the input points shapefile.
- `pass_field_name` Name of the attribute containing pass line ID.
- `output` Name of the output polygon shapefile.
- `width` Pass swath width (m).
- `max_change_in_heading` Max change in heading.
- `wd` Changes the working directory.
- `verbose_mode` Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
- `compress_rasters` Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- `command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.
wbt_yield_normalization

Yield normalization

Description

This tool can be used to normalize the yield points for a field.

Usage

```
wbt_yield_normalization(
  input,  
  yield_field,  
  output, 
  standardize = FALSE, 
  radius = NULL, 
  min_yield = 0, 
  max_yield = 99999.9, 
  wd = NULL, 
  verbose_mode = FALSE, 
  compress_rasters = FALSE, 
  command_only = FALSE 
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Name of the input points shapefile.</td>
</tr>
<tr>
<td>yield_field</td>
<td>Name of the attribute containing yield data.</td>
</tr>
<tr>
<td>output</td>
<td>Name of the output points shapefile.</td>
</tr>
<tr>
<td>standardize</td>
<td>Should the yield values be standardized (converted to z-scores) rather than normalized?</td>
</tr>
<tr>
<td>radius</td>
<td>Optional search radius, in metres. Only specify this value if you want to calculate locally normalized yield.</td>
</tr>
<tr>
<td>min_yield</td>
<td>Minimum yield value in output.</td>
</tr>
<tr>
<td>max_yield</td>
<td>Maximum yield value in output.</td>
</tr>
<tr>
<td>wd</td>
<td>Changes the working directory.</td>
</tr>
<tr>
<td>verbose_mode</td>
<td>Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.</td>
</tr>
<tr>
<td>compress_rasters</td>
<td>Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.</td>
</tr>
<tr>
<td>command_only</td>
<td>Return command that would be executed by system() rather than running tool.</td>
</tr>
</tbody>
</table>

Value

Returns the tool text outputs.
wbt_zlidar_to_las  Zlidar to las

Description

Converts one or more zlidar files into the LAS data format.

Usage

wbt_zlidar_to_las(
  inputs = NULL,
  outdir = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

Arguments

inputs  Input ZLidar files.
outdir  Output directory into which zlidar files are created. If unspecified, it is assumed to be the same as the inputs.
wd  Changes the working directory.
verbose_mode  Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters  Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only  Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.

wbt_zonal_statistics  Zonal statistics

Description

Extracts descriptive statistics for a group of patches in a raster.
Usage

```r
wbt_zonal_statistics(
  input,
  features,
  output = NULL,
  stat = "mean",
  out_table = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

Arguments

- **input**: Input data raster file.
- **features**: Input feature definition raster file.
- **output**: Output raster file.
- **stat**: Statistic to extract, including 'mean', 'median', 'minimum', 'maximum', 'range', 'standard deviation', and 'total'.
- **out_table**: Output HTML Table file.
- **wd**: Changes the working directory.
- **verbose_mode**: Sets verbose mode. If verbose mode is `FALSE`, tools will not print output messages.
- **compress_rasters**: Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
- **command_only**: Return command that would be executed by `system()` rather than running tool.

Value

Returns the tool text outputs.

---

**wbt_z_scores**

**Z scores**

Description

Standardizes the values in an input raster by converting to z-scores.
Usage

wbt_z_scores(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

Arguments

input Input raster file.
output Output raster file.
wd Changes the working directory.
verbose_mode Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters Sets the flag used by WhiteboxTools to determine whether to use compression for output rasters.
command_only Return command that would be executed by system() rather than running tool.

Value

Returns the tool text outputs.
Index

* datasets
  wbttoolparameters, 14
  wbttools, 15

check_whitebox_binary, 13

install_whitebox (wbt_install), 198
install_whitebox(), 196

sample_dem_data, 13

wbt_absolute_value, 15
wbt_accumulation_curvature, 16
wbt_adaptive_filter, 17
wbt_add, 18
wbt_add_point_coordinates_to_table, 19
wbt_aggregate_raster, 19
wbt_and, 20
wbt_anova, 21
wbt_arc_cos, 23
wbt_arc_sin, 24
wbt_arc_tan, 24
wbt_arccosh, 22
wbt_arcsinh, 25
wbt_arctanh, 26
wbt_ascii_to_las, 27
wbt_aspect, 28
wbt_assess_route, 29
wbt_atan2, 30
wbt_attribute_correlation, 31
wbt_attribute_correlation_neighbourhood_analysis, 31
wbt_attribute_histogram, 32
wbt_attribute_scattergram, 33
wbt_average_flowpath_slope, 34
wbt_average_normal_vector_angular_deviation, 35
wbt_average_overlay, 36
wbt_average_upslope_flowpath_length, 37
wbt_balance_contrast_enhancement, 37
wbt_basins, 38
wbt_bilateral_filter, 39
wbt_block_maximum_gridding, 40
wbt_block_minimum_gridding, 41
wbt_boundary_shape_complexity, 42
wbt_breach_depressions, 43
wbt_breach_depressions_least_cost, 44
wbt_breach_single_cell_pits, 45
wbt_buffer_raster, 45
wbt_burn_streams_at_roads, 46
wbt_canny_edge_detection, 47
wbt ceil, 48
wbt_centroid, 49
wbt_centroid_vector, 50
wbt_change_vector_analysis, 51
wbt_circular_variance_of_aspect, 52
wbt_classify_buildings_in_lidar, 53
wbt_classify_overlap_points, 54
wbt_clean_vector, 55
wbt clip, 55
wbt_clip_lidar_to_polygon, 56
wbt_clip_raster_to_polygon, 57
wbt closing, 58
wbt clump, 59
wbt compactness_ratio, 60
wbt_compress_rasters (wbt_init), 193
wbt conditional_evaluation, 60
wbt_conservative_smoothing_filter, 61
wbt_construct_vector_tin, 62
wbt contours_from_points, 63
wbt contours_from_raster, 64
wbt_convert_nodata_to_zero, 65
wbt_convert_raster_format, 66
wbt corner_detection, 67
wbt_correct_vignetting, 68
wbt cos, 69
wbt cosh, 69
wbt cost_allocation, 70

499
wbt_cost_distance, 71
wbt_cost_pathway, 72
wbt_count_if, 73
wbt_create_colour_composite, 74
wbt_create_hexagonal_vector_grid, 75
wbt_create_plane, 76
wbt_create_rectangular_vector_grid, 77
wbt_crispness_index, 78
wbt_cross_tabulation, 78
wbt_cumulative_distribution, 80
wbt_curvedness, 81
wbt_d8_flow_accumulation, 82
wbt_d8_mass_flux, 83
wbt_d8_pointer, 84
wbt_d_inf_flow_accumulation, 101
wbt_d_inf_mass_flux, 102
wbt_d_inf_pointer, 103
wbt_dbscan, 85
wbt_decrement, 86
wbt_default_path (wbt_init), 193
wbt_depth_in_sink, 86
wbt_dev_from_mean_elev, 87
wbt_diff_from_mean_elev, 90
wbt_diff_of_gaussian_filter, 91
wbt_difference, 88
wbt_difference_curvature, 89
wbt_direct_decorrelation_stretch, 93
wbt_directional_relief, 92
wbt_divide, 97
wbt_downslope_distance_to_stream, 98
wbt_downslope_flowpath_length, 99
wbt_downslope_index, 100
wbt_edge_contamination, 103
wbt_edge_density, 104
wbt_edge_preserving_mean_filter, 105
wbt_edge_proposition, 106
wbt_elev_above_pit, 109
wbt_elev_percentile, 109
wbt_elev_relative_to_min_max, 110
wbt_elev_relative_to_watershed_min_max, 111
wbt_elevation_above_stream, 107
wbt_elevation_above_stream_euclidean, 108
wbt_eliminate_coincident_points, 112
wbt_elongation_ratio, 113
wbt_embankment_mapping, 114
wbt_emboss_filter, 115
wbt_equal_to, 116
wbt_erase, 117
wbt_erase_polygon_from_lidar, 118
wbt_erase_polygon_from_raster, 119
wbt_euclidean_allocation, 120
wbt_euclidean_distance, 120
wbt_evulate_training_sites, 121
wbt_exe_path (wbt_init), 193
wbt_exe_path(), 13
wbt_exp, 122
wbt_exp2, 123
wbt_export_table_to_csv, 124
wbt_exposure_towards_wind_flux, 125
wbt_extend_vector_lines, 126
wbt_extract_nodes, 127
wbt_extract_raster_values_at_points, 127
wbt_extract_raster_values_at_points, 127
wbt_extract_streams, 128
wbt_extract_valleys, 129
wbt_fast_channel_head, 130
wbt_fast_almost_gaussian_filter, 131
wbt_fd8_flow_accumulation, 132
wbt_fd8_pointer, 133
wbt_feature_preserving_smoothing, 134
wbt_fetch_analysis, 135
wbt_fill_burn, 136
wbt_fill_depressions, 137
wbt_fill_depressions_planchon_and_darboux, 138
wbt_fill_depressions_wang_and_liu, 139
wbt_fill_missing_data, 140
wbt_fill_single_cell_pits, 141
wbt_filter_lidar_classes, 141
wbt_filter_lidar_scan_angles, 142
wbt_filter_raster_features_by_area, 143
wbt_find_flightline_edge_points, 144
wbt_find_flow_cells, 145
wbt_find_main_stem, 146
wbt_find_no_flow_cells, 147
wbt_find_parallel_flow, 147
wbt_find_patch_of_class_edges, 148
wbt_find_ridges, 149
INDEX

wbt_fix_dangling_arcs, 150
wbt_flatten_lakes, 151
wbt_flightline_overlap, 152
wbt_flip_image, 153
wbt_flood_order, 154
wbt_floor, 154
wbt_flow_accumulation_full_workflow, 155
wbt_flow_length_diff, 156
wbt_gamma_correction, 157
wbt_gaussian_contrast_stretch, 158
wbt_gaussian_curvature, 159
wbt_gaussian_filter, 160
wbt_gaussian_scale_space, 161
wbt_generalize_classified_raster, 162
wbt_generalize_with_similarity, 163
wbt_generating_function, 164
wbt_geomorphons, 165
wbt_greater_than, 166
wbt_hack_stream_order, 167
wbt_height_above_ground, 168
wbt_help, 168
wbt_high_pass_filter, 170
wbt_high_pass_median_filter, 171
wbt_highest_position, 169
wbt_hillshade, 172
wbt_hillslopes, 173
wbt_histogram_equalization, 174
wbt_histogram_matching, 175
wbt_histogram_matching_two_images, 176
wbt_hole_proportion, 177
wbt_horizon_angle, 178
wbt_horizontal_excess_curvature, 177
wbt_horton_stream_order, 179
wbt_hydrologic_connectivity, 180
wbt_hypsometric_analysis, 182
wbt_hypsometrically_tinted_hillshade, 181
wbt_idw_interpolation, 183
wbt_ihs_to_rgb, 184
wbt_image_autocorrelation, 185
wbt_image_correlation, 186
wbt_image_correlation_neighbourhood_analysis, 187
wbt_image_regression, 188
wbt_image_segmentation, 189
wbt_image_slider, 190
wbt_image_stack_profile, 191
wbt_impoundment_size_index, 192
wbt_in_place_add, 202
wbt_in_place_divide, 202
wbt_in_place_multiply, 203
wbt_in_place_subtract, 204
wbt_increment, 193
wbt_init, 193
wbt_insert_dams, 197
wbt_install, 198
wbt_integer_division, 198
wbt_integral_image, 199
wbt_intersect, 200
wbt_inverse_principal_component_analysis, 201
wbt_is_no_data, 206
wbt_isobasins, 205
wbt_jenson_snap_pour_points, 206
wbt_join_tables, 207
wbt_k_means_clustering, 212
wbt_k_nearest_mean_filter, 213
wbt_kappa_index, 208
wbt_knn_classification, 209
wbt_knn_regression, 210
wbt_ks_test_for_normality, 211
wbt_laplacian_filter, 214
wbt_laplacian_of_gaussian_filter, 215
wbt_las_to_asci, 216
wbt_las_to_laz, 217
wbt_las_to_multipoint_shapefile, 217
wbt_las_to_shapefile, 218
wbt_las_to_zlidar, 219
wbt_license, 224
wbt_lidar_block_maximum, 225
wbt_lidar_block_minimum, 226
wbt_lidar_classify_subset, 227
wbt_lidar_colourize, 228
wbt_lidar_contour, 229
wbt_lidar_digital_surface_model, 230
wbt_lidar_elevation_slice, 231
wbt_lidar_ground_point_filter, 232
wbt_lidar_hex_binning, 233
wbt_lidar_histogram, 234
wbt_lidar_histogram, 235
wbt_multiscale_std_dev_normals_signature, 319
wbt_multiscale_topographic_position_image, 320
wbt_narrowness_index, 322
wbt_natural_neighbour_interpolation, 322
wbt_nearest_neighbour_gridding, 323
wbt_negate, 324
wbt_new_raster_from_base, 325
wbt_normal_vectors, 327
wbt_normalized_difference_index, 326
wbt_not, 328
wbt_not_equal_to, 329
wbt_num_downslope_neighbours, 330
wbt_num_inflowing_neighbours, 330
wbt_num_upslope_neighbours, 331
wbt_olympic_filter, 332
wbt_opening, 333
wbt_openness, 334
wbt_options (wbt_init), 193
wbt_or, 335
wbt_paired_sample_t_test, 336
wbt_panchromatic_sharpening, 337
wbt_parallelepiped_classification, 338
wbt_patch_orientation, 339
wbt_pennock_landform_class, 339
wbt_percent_elev_range, 342
wbt_percent_equal_to, 343
wbt_percent_greater_than, 344
wbt_percent_less_than, 345
wbt_percentage_contrast_stretch, 340
wbt_percentile_filter, 341
wbt_perimeter_area_ratio, 346
wbt_phi_coefficient, 347
wbt_pick_from_list, 348
wbt_plan_curvature, 349
wbt_polygon_area, 351
wbt_polygon_long_axis, 352
wbt_polygon_perimeter, 353
wbt_polygon_short_axis, 353
wbt_polygonize, 350
wbt_polygons_to_lines, 350
wbt_power, 354
wbt_prewitt_filter, 355
wbt_principal_component_analysis, 356
wbt_print_geo_tiff_tags, 357
wbt_profile, 357
wbt_profile_curvature, 358
wbt_qin_flow_accumulation, 359
wbt_quantiles, 360
wbt_quinn_flow_accumulation, 361
wbt_radial_basis_function_interpolation, 362
wbt_radius_of_gyration, 363
wbt_raise_walls, 364
wbt_random_field, 365
wbt_random_forest_classification, 366
wbt_random_forest_regression, 367
wbt_random_sample, 368
wbt_range_filter, 369
wbt_raster_area, 371
wbt_raster_calculator, 372
wbt_raster_cell_assignment, 373
wbt_raster_histogram, 374
wbt_raster_perimeter, 375
wbt_raster_streams_to_vector, 376
wbt_raster_summary_stats, 377
wbt_raster_to_vector_lines, 377
wbt_raster_to_vector_points, 378
wbt_raster_to_vector_polygons, 379
wbt_rasterize_streams, 370
wbt_reciprocal, 380
wbt_reclass, 381
wbt_reclass_equal_interval, 382
wbt_reclass_from_file, 383
wbt_reconcile_multiple_headers, 384
wbt_recreate_pass_lines, 385
wbt_reinitialize_attribute_table, 386
wbt_related_circumscribing_circle, 387
wbt_relative_aspect, 387
wbt_relative_topographic_position, 388
wbt_remove_field_edge_points, 389
wbt_remove_off_terrain_objects, 390
wbt_remove_polygon_holes, 391
wbt_remove_short_streams, 392
wbt_remove_spurs, 393
wbt_repair_stream_vector_topology, 394
wbt_resample, 395
wbt_rescale_value_range, 396
wbt_rgb_to_ihs, 397
wbt_rho8_flow_accumulation, 398
wbt_rho8_pointer, 399
wbt_ring_curvature, 400
wbt_roberts_cross_filter, 401
wbt_root_mean_square_error, 402
wbt_rotor, 402
wbt_round, 403
wbt_ruggedness_index, 404
wbt_run_tool, 405
wbt_scharr_filter, 406
wbt_sediment_transport_index, 407
wbt_select_tiles_by_polygon, 408
wbt_set_nodata_value, 409
wbt_shadow_animation, 410
wbt_shadow_image, 411
wbt_shape_complexity_index, 412
wbt_shape_complexity_index_raster, 413
wbt_shreve_stream_magnitude, 414
wbt_sigmoidal_contrast_stretch, 415
wbt_sin, 416
wbt_single_part_to_multi_part, 417
wbt_sinh, 418
wbt_sink, 418
wbt_slope, 419
wbt_slope_vs_aspect_plot, 420
wbt_slope_vs_elevation_plot, 421
wbt_smooth_vectors, 422
wbt_smooth_vegetation_residual, 423
wbt_snap_pour_points, 424
wbt_sobel_filter, 425
wbt_spherical_std_dev_of_normals, 426
wbt_split_colour_composite, 427
wbt_split_vector_lines, 428
wbt_split_with_lines, 429
wbt_square, 430
wbt_square_root, 430
wbt_standard_deviation_contrast_stretch, 431
wbt_standard_deviation_filter, 432
wbt_standard_deviation_of_slope, 433
wbt_stochastic_depression_analysis, 434
wbt_strahler_order_basins, 435
wbt_strahler_stream_order, 436
wbt_stream_link_class, 437
wbt_stream_link_identifier, 438
wbt_stream_link_length, 439
wbt_stream_link_slope, 440
wbt_stream_power_index, 441
wbt_stream_slope_continuous, 442
wbt_subbasins, 443
wbt_subtract, 444
wbt_sum_overlay, 445
wbt_surface_area_ratio, 445
wbt_svm_classification, 446
wbt_svm_regression, 447
wbt_symmetrical_difference, 449
wbt_tan, 450
wbt_tangential_curvature, 450
wbt_tanh, 451
wbt_thicken_raster_line, 452
wbt_time_in_daylight, 453
wbt_tin_gridding, 454
wbt_to_degrees, 462
wbt_to_radians, 462
wbt_tool_help, 456
wbt_tool_parameters, 456
wbt_tool_parameters(), 15
wbt_toolbox, 455
wbt_tophat_transform, 457
wbt_topographic_position_animation, 458
wbt_topological_stream_order, 459
wbt_total_curvature, 460
wbt_total_filter, 461
wbt_trace_downslope_flowpaths, 463
wbt_trend_surface, 464
wbt_trend_surface_vector_points, 465
wbt_tributary_identifier, 466
wbt_truncate, 467
wbt_turning_bands_simulation, 468
wbt_two_sample_ks_test, 469
wbt_union, 470
wbt_unnest_basins, 471
wbt_unsharp_masking, 472
wbt_unsphericity, 473
wbt_update_nodata_cells, 474
wbt_upslope_depression_storage, 475
wbt_user_defined_weights_filter, 475
wbt_vector_hex_binning, 476
wbt_vector_lines_to_raster, 477
wbt_vector_points_to_raster, 478
wbt_vector_polygons_to_raster, 479
wbt_vector_stream_network_analysis, 480
wbtVerbose (wbt_init), 193
wbt_version, 481
wbt_vertical_excess_curvature, 482
wbt_view_code, 484
wbt_viewshed, 483
wbt_visibility_index, 484
wbt_voronoi_diagram, 485
wbt_watershed, 486
wbt_wd (wbt_init), 193
wbt_weighted_overlay, 487
wbt_weighted_sum, 488
wbt_wetness_index, 489
wbt_wilcoxon_signed_rank_test, 490
wbt_write_function_memory_insertion, 491
wbt_xor, 492
wbt_yield_filter, 493
wbt_yield_map, 494
wbt_yield_normalization, 495
wbt_z_scores, 497
wbt_zlidar_to_las, 496
wbt_zonal_statistics, 496
wbttoolparameters, 14
wbttools, 15, 15
whitebox, 196