Package ‘ripa’

February 20, 2015

Version 2.0-2
Date 2014-05-29
Title R Image Processing and Analysis
Maintainer Talita Perciano <talitaperciano@gmail.com>
Depends R (>= 2.8.1), tcltk, parallel
Suggests e1071, rggobi, reshape, methods, jpeg, png, tkrplot, fftw,
foreach, doSNOW
Enhances doMC
SystemRequirements BWidget, Tktable, Img, libjpeg
Description A package including various functions for image processing and analysis. With this pack-
age is possible to process and analyse RGB, LAN (multispectral) and AVIRIS (hyperspec-
tral) images. This packages also provides functions for reading JPEG files, ex-
tracted from the archived 'rimage' package.
License GPL (>= 2) | file LICENSE
Imports Rcpp (>= 0.11.0)
LinkingTo Rcpp
URL http://www.r-project.org
Author Talita Perciano [aut, cre],
Alejandro C Frery [ctb]
NeedsCompilation yes
Repository CRAN
Date/Publication 2014-05-31 01:32:57

R topics documented:

  ripa-package ................................................................. 3
  aviris_band-class ............................................................ 3
  aviris_image-class ........................................................... 4
  aviris_scene-class ............................................................ 5
  aviris_training-class ......................................................... 6
R topics documented:

- cgauss .......................................................... 7
- checkTab ..................................................... 7
- clinear ....................................................... 8
- clipping ...................................................... 8
- contBriImg .................................................. 9
- contrast ...................................................... 10
- fftImg ....................................................... 10
- fftw .......................................................... 11
- Grey .......................................................... 12
- highpass ..................................................... 12
- imagematrix ................................................ 13
- imageType .................................................. 14
- initialize-methods ........................................ 15
- lband ......................................................... 15
- lbandsample ................................................ 16
- limage ........................................................ 16
- loadBand .................................................... 17
- loadBandSample ........................................... 17
- logo ........................................................... 18
- lowpass ...................................................... 18
- lscene ........................................................ 19
- medianImg .................................................. 20
- modalDialog ............................................... 20
- normalize ................................................... 21
- plot.imagematrix .......................................... 21
- plot_band.aviris_band .................................... 22
- print.imagematrix ......................................... 23
- print_information.aviris_band ......................... 23
- print_information.aviris_image ....................... 24
- print_information.aviris_scene ....................... 24
- print_information.aviris_training .................... 25
- read.aviris ................................................ 25
- read.lan ..................................................... 26
- RGB .......................................................... 26
- rgb2grey .................................................... 27
- ripaEnv ..................................................... 27
- RIPAgui ..................................................... 28
- stretchImg ................................................ 28
- takeSamples ............................................... 29
- wband ........................................................ 29
- write.lan .................................................... 30
- writeBand .................................................. 30
- zoom .......................................................... 31
- zoomGrey ................................................... 31
- zoomRGB .................................................... 32
- Zprofile ..................................................... 32

Index .......................................................... 34
Description

A package including various functions for image processing and analysis. With this package is possible to process and analyse RGB, LAN (multispectral) and AVIRIS (hyperspectral) images. This package also provides functions for reading JPEG files, extracted from the archived 'rimage' package.

Details

Package: ripa
Type: Package
Version: 2.0-1
Date: 2014-05-12
License: GPL version 2 or newer

Author(s)

Talita Perciano and Alejandro C Frery
Maintainer: Talita Perciano <talitapericiao@gmail.com>

aviris_band-class

Description

Class that represents a band of an AVIRIS image

Objects from the Class

Objects can be created by calls of the form new("aviris_band", scene, band, type, numberOfLines, samples, data)

Slots

scene: Object of class "character". Name of the image scene.
band: Object of class "numeric". Number of the band.
type: Object of class "character". Type of the band, like "reflectance" for instance.
numberOfLines: Object of class "numeric".
samples: Object of class "numeric".
data: Object of class "matrix".
min: Object of class "numeric".
max: Object of class "numeric".
mean: Object of class "numeric".
sd: Object of class "numeric".

Methods

initialize signature(.Object = "aviris_band"): ...

Author(s)

Marcelo Almiron and Adrian Muract

See Also

See Also aviris_image-class, aviris_scene-class, aviris_training-class

Examples

showClass("aviris_band")
**aviris_scene-class**

**Author(s)**
Marcelo Almiron and Adrian Muract

**See Also**
See Also `aviris_band-class`, `aviris_scene-class`, `aviris_training-class`

**Examples**
```
showClass("aviris_image")
```

---

**Description**
Class that represents a scene of an AVIRIS image

**Objects from the Class**
Objects can be created by calls of the form `new("aviris_scene", name, numberOfLines, samples, bands, imageName, type, path)`.

**Slots**
- `name`: Object of class "character".
- `numberOfLines`: Object of class "numeric".
- `samples`: Object of class "numeric".
- `bands`: Object of class "numeric".
- `imageName`: Object of class "character".
- `type`: Object of class "character".
- `path`: Object of class "character".

**Methods**
- `initialize` signature(.Object = "aviris_scene"): ...

**Author(s)**
Marcelo Almiron and Adrian Muract

**See Also**
See Also `aviris_image-class`, `aviris_band-class`, `aviris_training-class`

**Examples**
```
showClass("aviris_scene")
```
**aviris_training-class**  
Class "aviris_training"

Description

Class to create a training category.

Objects from the Class

Objects can be created by calls of the form `new("aviris_training", category, color, scene, band, posX, posY)`.

Slots

category: Object of class "character". Name of the new category.
color: Object of class "character". Color that will represent the category.
scene: Object of class "aviris_scene".
bands: Object of class "list".
posX: Object of class "numeric".
posY: Object of class "numeric".

Methods

initialize signature(.Object = "aviris_training"): ...

Author(s)

Marcelo Almiron and Adrian Muract

See Also

See Also `aviris_image-class`, `aviris_scene-class`, `aviris_training-class`

Examples

`showClass("aviris_training")`
cgauss

Contrast Gaussian expansion algorithm for AVIRIS images

Description
Applies the contrast Gaussian expansion algorithm to the input data.

Usage
cgauss(Z)

Arguments
Z Input data as a matrix or vector

Value
Returns the expansion result

Author(s)
Marcelo Almiron and Adrian Muract

See Also
See Also clineal

checkTab

Function to check active tab

Description
Checks the active tab of the GUI.

Usage
checkTab()

Author(s)
Talita Perciano


clineal  

*Contrast linear expansion algorithm for AVIRIS images*

**Description**

Applies the contrast linear expansion algorithm to the input data.

**Usage**

clineal(Z, A, B)

**Arguments**

- **Z**: Input data as a matrix or vector
- **A**: Minimum value of range (0)
- **B**: Maximum value of range (1)

**Value**

Returns the expansion result.

**Author(s)**

Marcelo Almiron and Adrian Muract

**See Also**

See Also `cgauss`

---

clipping  

*Clipping image*

**Description**

This function returns the image which restricts pixel value from the specified lowest value to the specified highest value in the original image. This means that the pixels which have lower value than the given lowest (default: 0) are replaced to the lowest and the pixels have greater value than the given highest (default: 1) are replaced to the highest.

**Usage**

clipping(img, low=0, high=1)
**contBriImg**

**Arguments**

- `img`: target image
- `low`: lowest value
- `high`: highest value

**Value**

Data of the same mode as `img`

**Examples**

```r
data(logo)
op <- par(mfrow=c(2,2))
plot(logo, main="Source Image")

# the appearance of next one doesn't change because of normalization.
plot(normalize(2*logo), main="Doubled pixel value with normalization")

# the next one is saturated as expected
plot(clipping(2*logo), main="Doubled pixel value with clipping")
```

---

## Description

Applies values of contrast and brightness on an image.

## Usage

```r
contBriImg(img, cont, bri)
```

## Arguments

- `img`: Input image
- `cont`: New value for contrast (0.0 to 1.0)
- `bri`: New value for brightness (-1.0 to 1.0)

## Value

Return the result image after applying the values of contrast and brightness

## Author(s)

Talita Perciano
contrast

Interface to choose the contrast expansion type

Description

Chooses the type of contrast expansion to apply on band.

Usage

contrast(band, type = c("gauss", "lineal"), ...)

Arguments

- band: One band of the image (matrix)
- type: Type of the contrast expansion
- ...: Other possible arguments

Author(s)

Marcelo Almiron and Adrian Muract

See Also

See Also `cgauss`, `clineal`

fftImg

Compute FFT image

Description

This function computates the power spectrum of a given image by FFT.

Usage

fftImg(img)

Arguments

- img: target image

Value

an imagematrix
fftw

References

FFTW (Fastest Fourier Transform in the West) http://www.fftw.org

See Also

fftw, imagematrix

Examples

```r
## Not run:
data(logo)
plot(normalize(fftImg(logo)))

## End(Not run)
```

fftw Apply FFT to 2-Dimensional Data

Description

This function applies FFT to 2-dimensional data (i.e. image) using fftw library.

Usage

`fftw(img, dir = -1, debug=FALSE)`

Arguments

- `img`: target image
- `dir`: set -1 for normal transformation and 1 for inverse transformation
- `debug`: set TRUE if you want to output debug message

Value

a matrix of complex number

References

FFTW (Fastest Fourier Transform in the West) http://www.fftw.org

See Also

`fftw`
Description

Makes an interface for gray scale view of an image band.

Usage

Grey(band, x0, y0, ...)

Arguments

band One band of the image (matrix)
x0 1
y0 1
... Other possible arguments

Author(s)

Marcelo Almiron and Adrian Muract

See Also

See Also plot_band.aviris_band, RGB

highpass

High pass filter for image

Description

Computes a high-pass filtered image with dimensions of the given input image. the hp pasing distance is given via radius.

Usage

highpass(img, radius)

Arguments

img pixmap or image
radius gives the blocking radius
imagematrix

Value

an imagematrix

See Also

lowpass

Examples

```r
## Not run:
data/logo
plot(normalize(highpass/logo))
## End(Not run)
```

### Description

This function makes an imagematrix object from a matrix. This data structure is primary data structure to represent image in rimage package.

### Usage

```r
imagematrix(mat, type=NULL,
             ncol=dim(mat)[1], nrow=dim(mat)[2], noclipping=FALSE)
```

### Arguments

- `mat`: array, matrix or vector
- `type`: "rgb" or "grey"
- `ncol`: width of image
- `nrow`: height of image
- `noclipping`: TRUE if you disable automatic clipping. See details.

### Details

For grey scale image, matrix should be given in the form of 2 dimensional matrix. First dimension is row, and second dimension is column.

For rgb image, matrix should be given in the form of 3 dimensional array (row, column, channel). `mat[,1]`, `mat[,2]`, `mat[,3]` are red plane, green plane and blue plane, respectively.

You can omit 'type' specification if you give a proper array or matrix. Also, if you give a rgb image matrix and specify "grey" as type, the rgb image matrix is automatically converted to a grey scale image.
This function automatically clips the pixel values which are less than 0 or greater than 1. If you want to disable this behavior, give 'noclipiing=TRUE'.

The major difference between imagematrix and pixmap is representation method. pixmap (>0.3) uses OOP class. On the other hand, rimage uses traditional S class. The advantage of traditional S class in representing image is that one can deal with the data structure as an ordinary matrix.

The minor difference between imagematrix and pixmap is automatic data conversion behavior. pixmap normalizes a given matrix automatically if any element of the matrix is out of range between 0 and 1. On the other hand, imagematrix clips the matrix, which means that the pixels which have lower value than 0 are replaced to 0 and the pixels have greater value than 1 are replaced to 1.

Value

return an imagematrix object

See Also

plot.imagematrix, print.imagematrix

Examples

p <- q <- seq(-1, 1, length=20)
r <- 1 - outer(p^2, q^2, "+ ") / 2
plot(imagematrix(r))
Examples

```r
## Not run:
x <- read.jpeg(system.file("extdata", "Rlogo.jpg", package="ripa"))
cat("Image Type", imageType(x))

x.grey <- rgb2grey(x)
cat("Image Type", imageType(x grey))

## End(Not run)
```

initialize-methods

Methods for Function initialize

Methods

- **.Object = "aviris_band"** Create an AVIRIS band.
- **.Object = "aviris_image"** Create an AVIRIS image.
- **.Object = "aviris_scene"** Create an AVIRIS scene.
- **.Object = "aviris_training"** Create a new category for training.

lband

Interface to load an AVIRIS image band

Description

Loads an image band of a scene.

Usage

`lband(scene, b)`

Arguments

- `scene` An object of type scene
- `b` Number of the band

Author(s)

Marcelo Almiron and Adrian Muract

See Also

See Also `lbandsample`
lbandsample  
*Interface to load an AVIRIS image band sample*

**Description**

Loads an image band sample of a scene.

**Usage**

```
lbandsample(scene, b)
```

**Arguments**

- `scene`: An object of type scene
- `b`: Number of the band

**Author(s)**

Marcelo Almiron and Adrian Muract

**See Also**

See Also as `lband`.

limage  
*Interface to load the header of an AVIRIS image*

**Description**

Loads the header of an image.

**Usage**

```
limage(H, type)
```

**Arguments**

- `H`: Name of the header
- `type`: Type of the image ("reflectance" or "radiance")

**Author(s)**

Marcelo Almiron and Adrian Muract
loadBand

Function to load an AVIRIS image band

Description

Loads an image band using 1band interface.

Usage

loadBand(I, X = 5, C = 614, F = 512, B = 224)

Arguments

<table>
<thead>
<tr>
<th>I</th>
<th>Name of the image file</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Number of the band</td>
</tr>
<tr>
<td>C</td>
<td>Number of columns of each band</td>
</tr>
<tr>
<td>F</td>
<td>Number of lines of each band</td>
</tr>
<tr>
<td>B</td>
<td>Total number of bands</td>
</tr>
</tbody>
</table>

Author(s)

Marcelo Almiron and Adrian Muract

See Also

See Also as loadBandSample

loadBandSample

Function to load an AVIRIS image band sample

Description

Loads an image band sample using 1bandsample interface.

Usage

loadBandSample(I, X = 5, C = 614, F = 30, B = 224)

Arguments

<table>
<thead>
<tr>
<th>I</th>
<th>Name of the image file</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Number of the band</td>
</tr>
<tr>
<td>C</td>
<td>Number of columns of each band</td>
</tr>
<tr>
<td>F</td>
<td>Number of lines of each band</td>
</tr>
<tr>
<td>B</td>
<td>Total number of bands</td>
</tr>
</tbody>
</table>
Author(s)
Marcelo Almiron and Adrian Muract

See Also
See Also as `loadBand`.

---

### logo

*R logo imagematrix*

---

**Description**
The imagematrix object of R logo of the size 101x77.

**Usage**
```r
data(logo)
```

**Format**
imagematrix

**Examples**
```r
data(logo)
plot(logo)
```

---

### lowpass

*Low Pass Filter for Image*

---

**Description**
Computes a low-pass filtered image with dimensions of the given input image. The lowpassing distance is given via `radius`.

**Usage**
```r
lowpass(img, radius)
```

**Arguments**
```r
  img  pixmap or matrix
  radius  gives the pass radius
```
Interface to load an AVIRIS image scene

Description

Load an AVIRIS image scene.

Usage

lscene(image, n)

Arguments

image Name of the file
n Number of the scene

Author(s)

Marcelo Almiron and Adrian Muract
**medianImg**  
*Median filter*

**Description**
Applies the median filter on an image

**Usage**
```python
cmedianImg(img, mask)
```

**Arguments**
- `img`  The input image as a matrix
- `mask`  The mask length

**Author(s)**
Talita Perciano

---

**modalDialog**  
*Modal dialog*

**Description**
Builds a tcltk modal dialog.

**Usage**
```python
modalDialog(title, question, entryInit, entryWidth = 20, returnValOnCancel = "ID_CANCEL")
```

**Arguments**
- `title`  Title of the dialog
- `question`  Question of the dialog
- `entryInit`  The initial entry value
- `entryWidth`  The entry width
- `returnValOnCancel`  Value to be returned on cancel

**Author(s)**
Talita Perciano
normalize

Normalization for vector and matrix

Description

This function normalizes image so that the minimum value is 0 and the maximum value is 1.

Usage

normalize(img)

Arguments

img
target image

Value

Data of the same mode as 'img', in which minimum value is 0 and maximum value is 1.

Examples

data(logo)
plot(normalize(logo))

plot.imagematrix

Plotting an imagematrix object

Description

This function outputs an imagematrix object as an image.

Usage

## S3 method for class 'imagematrix'
plot(x, ...)

Arguments

x
target image
...
plotting options

See Also

imagematrix
Examples

```r
op <- par(mfrow=c(1,2))

data(logo)
plot(logo, main="plot(logo)")
plot(logo^2, main="plot(logo^2)")

par(op)
```

---

Function to plot an AVIRIS band

Description

Plots an AVIRIS band defined as a composition of three bands (R, G and B)

Usage

```r
## S3 method for class 'aviris_band'
plot_band(R = NULL, G = NULL, B = NULL, type = NULL, x0 = 1, y0 = 1, ...)
```

Arguments

- **R**
  - The R band
- **G**
  - The G band
- **B**
  - The B band
- **type**
  - Type of the image: "grey" or "rgb".
- **x0**
  - 1
- **y0**
  - 1
- ...
  - Other possible arguments

Author(s)

Marcelo Almiron and Adrian Muract
print.imagematrix

Print information on a given imagematrix object

Description

This function outputs information on a given imagematrix object.

Usage

```r
## S3 method for class 'imagematrix'
print(x, ...)
```

Arguments

- `x` target image
- `...` ignored (dummy)

See Also

- `imagematrix`

Examples

```r
data(logo)
print(logo)
```

print_information.aviris_band

Function to print an AVIRIS band

Description

Prints information about an AVIRIS band.

Usage

```r
## S3 method for class 'aviris_band'
print_information(Object)
```

Arguments

- `Object` An object of class aviris_band.

Author(s)

Marcelo Almiron and Adrian Muract
print_information.aviris_image

Function to print an AVIRIS image

Description
Prints information about an AVIRIS image.

Usage

```
## S3 method for class 'aviris_image'
print_information(Object)
```

Arguments

Object An object of class aviris_image.

Author(s)
Marcelo Almiron and Adrian Muract

print_information.aviris_scene

Function to print an AVIRIS scene

Description
Prints information about an AVIRIS scene.

Usage

```
## S3 method for class 'aviris_scene'
print_information(Object)
```

Arguments

Object An object of class aviris_scene.

Author(s)
Marcelo Almiron and Adrian Muract
print_information.aviris_training

Function to print an AVIRIS training category

Description
Prints information about an AVIRIS training category.

Usage
## S3 method for class 'aviris_training'
print_information(Object)

Arguments

Object An object of class aviris_training.

Author(s)
Marcelo Almiron and Adrian Muract

read.aviris

Function to read an AVIRIS image

Description
Reads an AVIRIS image.

Usage
read.aviris(fileName, bandsindexes, bands_local, use_parallel)

Arguments

fileName The name of the file
bandsindexes Indexes of the bands to be read
bands_local Local variable to store the bands
use_parallel Indicates if the function should be run in parallel or not (1=yes, 0=no)

Author(s)
Talita Perciano

See Also
See Also as read.lan, ~~~
read.lan  
*Function to read a LAN image*

**Description**
Reads a LAN image.

**Usage**
read.lan(arquivo)

**Arguments**
arquivo  
The name of the file

**Author(s)**
Talita Perciano

---

RGB  
*Interface for RGB view for AVIRIS images*

**Description**
Makes an interface for RGB view of an image band.

**Usage**
RGB(red, green, blue, x0, y0, ...)

**Arguments**
red  
The R band
green  
The G band
blue  
The B band
x0  
1
y0  
1
...  
Other possible arguments

**Author(s)**
Marcelo Almiron and Adrian Muract

**See Also**
See Also as Grey.
rgb2grey

Convert color imagematrix to grey imagematrix

Description

This function convert color imagematrix to grey imagematrix.

Usage

rgb2grey(img, coefs=c(0.30, 0.59, 0.11))

Arguments

- **img**: target image
- **coefs**: coefficients for red plane, green plane, and blue plane.

Value

grey imagematrix

Examples

```r
## Not run:
x <- read.jpeg(system.file("extdata", "Rlogo.jpg", package="ripa"))
plot(rgb2grey(x))

## End(Not run)
```

ripaEnv

Environment for package ripa

Description

Environment with variables used by the package
RIPAgui

*Function to build the ripa package GUI*

**Description**

Builds the GUI for ripa package. After loading the package, users should run this function to use all the developed tools.

**Usage**

RIPAgui()

**Author(s)**

Talita Perciano

---

stretchImg

*Function to apply contrast linear stretch*

**Description**

Applies contrast linear stretch to an image.

**Usage**

stretchImg(img)

**Arguments**

- **img** Input image as a matrix

**Author(s)**

Talita Perciano
takeSamples

Function to take training samples for AVIRIS images

Description
Take samples for future training.

Usage
takeSamples(t, n = NULL, Sample = NULL, Line = NULL)

Arguments
t Training class
n Number of samples
Sample Column of the image
Line Line of the image

Author(s)
Marcelo Almiron and Adrian Muract

wband

Interface to save an AVIRIS image band

Description
Saves an image band of a scene.

Usage
wband(scene, band)

Arguments
scene An object of type scene
band Number of the band

Author(s)
Marcelo Almiron and Adrian Muract

See Also
See Also as lband.
write.band

Function to save LAN images

Description
Saves LAN images.

Usage
write.band(arquivo, img)

Arguments
- arquivo: Name of the file
- img: Data to be saved

Author(s)
Talita Perciano

See Also
See Also read.band

writeBand

Function to save an AVIRIS image band

Description
Saves an image band using wband interface.

Usage
writeBand(I, Z, X = NA, C = 614, F = 512, B = 224)

Arguments
- I: Name of the image file
- Z: Data to be saved
- X: Number of the band
- C: Number of columns of each band
- F: Number of lines of each band
- B: Total number of bands
**zoom**

*Interface for zoomGrey and zoomRGB*

**Description**

Interface for zoomGrey and zoomRGB.

**Usage**

```r
zoom(R = NULL, G = NULL, B = NULL)
```

**Arguments**

- `R` The R band
- `G` The G band
- `B` The B band

**Author(s)**

Marcelo Almiron and Adrian Muract

**See Also**

See Also `loadBand`

---

**zoomGrey**

*Function to apply zoom to grey images*

**Description**

Applies zoom to an grey image.

**Usage**

```r
zoomGrey(band)
```

**Arguments**

- `band` Data input
Author(s)
Marcelo Almiron and Adrian Muract

See Also
See Also zoomRGB

---

**zoomRGB**

Function to apply zoom to RGB images

**Description**
Applies zoom to a RGB image.

**Usage**
zoomRGB(Red, Green, Blue)

**Arguments**
- Red  The R band
- Green The G band
- Blue  The B band

Author(s)
Marcelo Almiron and Adrian Muract

See Also
See Also zoomGrey

---

**Zprofile**

Function for Z Profile

**Description**
Show the Z Profile of a pixel.

**Usage**
Zprofile(scene, X = NULL, Y = NULL)
Arguments

<table>
<thead>
<tr>
<th>scene</th>
<th>Scene</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X-coordinate</td>
</tr>
<tr>
<td>Y</td>
<td>Y-coordinate</td>
</tr>
</tbody>
</table>

Author(s)

Marcelo Almiron and Adrian Muract
Index

*Topic classes
  aviris_band-class, 3
  aviris_image-class, 4
  aviris_scene-class, 5
  aviris_training-class, 6

*Topic datasets
  logo, 18

*Topic methods
  cgauss, 7
  checkTab, 7
  clineal, 8
  contBriImg, 9
  contrast, 10
  Grey, 12
  initialize-methods, 15
  lband, 15
  lbandsample, 16
  limage, 16
  loadBand, 17
  loadBandSample, 17
  lscene, 19
  medianImg, 20
  modalDialog, 20
  plot_band.aviris_band, 22
  print_information.aviris_band, 23
  print_information.aviris_image, 24
  print_information.aviris_scene, 24
  print_information.aviris_training, 25
  read.aviris, 25
  read.lan, 26
  RGB, 26
  RIPAgui, 28
  stretchImg, 28
  takeSamples, 29
  wband, 29
  write.lan, 30
  writeBand, 30
  zoom, 31
  zoomGrey, 31
  zoomRGB, 32
  Zprofile, 32

*Topic misc
  clipping, 8
  fftImg, 10
  fftw, 11
  highpass, 12
  imagematrix, 13
  imageType, 14
  lowpass, 18
  normalize, 21
  plot.imagematrix, 21
  print.imagematrix, 23
  rgb2grey, 27

*Topic package
  ripa-package, 3
  aviris_band-class, 3
  aviris_band-method (aviris_band-class), 3
  aviris_image-class, 4
  aviris_image-method (aviris_image-class), 4
  aviris_scene-class, 5
  aviris_scene-method (aviris_scene-class), 5
  aviris_training-class, 6
  aviris_training-method (aviris_training-class), 6
  cgauss, 7, 8, 10
  checkTab, 7
  clineal, 7, 8, 10
  clipping, 8
  contBriImg, 9
  contrast, 10
  fftImg, 10
  fftw, 11, 11
INDEX

Grey, 12, 26

highpass, 12, 19

imagematrix, 11, 13, 21, 23
imageType, 14
initialize, aviris_band-method
(initialize-methods), 15
initialize, aviris_image-method
(initialize-methods), 15
initialize, aviris_scene-method
(initialize-methods), 15
initialize, aviris_training-method
(initialize-methods), 15
initialize-methods, 15

lband, 15, 16, 29
lbandsample, 15, 16
limage, 16
loadBand, 17, 18, 31
loadBandSample, 17, 17
logo, 18
lowpass, 13, 18
lscene, 19

medianImg, 20
modalDialog, 20

normalize, 21

plot.imagematrix, 14, 21
plot_band.aviris_band, 12, 22
print.imagematrix, 14, 23
print_information.aviris_band, 23
print_information.aviris_image, 24
print_information.aviris_scene, 24
print_information.aviris_training, 25

read.aviris, 25
read.lan, 25, 26, 30
RGB, 12, 26
rgb2grey, 27
ripa (ripa-package), 3
ripa-package, 3
ripaEnv, 27
RIPAgui, 28

stretchImg, 28

takeSamples, 29

wband, 29
write.lan, 30
writeBand, 30

zoom, 31
zoomGrey, 31, 31, 32
zoomRGB, 31, 32, 32
Zprofile, 32