

Package ‘rtiff’

November 25, 2019

Type Package

Title Read and Write TIFF Files

Version 1.4.7

Date 2019-11-21

Depends pixmap

Author Eric Kort <eric.kort@vai.org>

Maintainer Eric Kort <eric.kort@vai.org>

SystemRequirements libtiff

Description Reads and writes TIFF format images and returns them as a pixmap object. Because the resulting object can be very large for even modestly sized TIFF images, images can be reduced as they are read for improved performance. This package is a wrapper around libtiff (www.libtiff.org), on which it depends (i.e. the libtiff shared library must be on your PATH for the binary to work, and tiffio.h must be on your system to build the package from source). By using libtiff's highlevel TIFFReadRGBAImage function, this package inherently supports a wide range of image formats and compression schemes. This package also provides an implementation of the Ridler Autothresholding algorithm for easy generation of binary masks as described in Ridler & Calvard (1978) <[doi:10.1109/TSMC.1978.4310039](https://doi.org/10.1109/TSMC.1978.4310039)>.

License GPL-3

Repository CRAN

Date/Publication 2019-11-25 20:40:02 UTC

NeedsCompilation yes

R topics documented:

rtiff-package	2
autoThreshold	3
getDescription	4
newPixmapRGB	5

plot.matrix	6
readTiff	6
updateDescription	8
writeTiff	8

Index	10
--------------	-----------

rtiff-package	<i>A tiff reader for R.</i>
---------------	-----------------------------

Description

This package will read TIFF format images and return them as a pixmap object. Because the resulting object can be very large for even modestly sized TIFF images, images can be reduced as they are read for improved performance. This package is a wrapper around libtiff (www.libtiff.org), on which it depends. By using libtiff's highlevel TIFFReadRGBAImage function, this package inherently support a wide range of image formats and compression schemes. This package also provides an implementation of the Ridler Autothresholding algorithm for easy generation of binary masks.

Details

```
Package:  rtiff
Type:     Package
Version:  1.0
Date:     2005-12-20
License:  GPL
```

A tiff image can be loaded using readTiff, and then plotted, or processed. Regardless of the color-space of the original image, this function returns an RGB pixmap. The image rasters can be accessed through the red, green, and blue slots of the pixmap object returned. These rasters may be further processed using any of the widely described image analysis algorithms. As a first step, and autoThreshold function is provided by this package to create binary masks of pixmap channels of interest.

Author(s)

Eric Kort <eric.kort@vai.org> Maintainer: Eric Kort <eric.kort@vai.org>

References

www.libtiff.org

Examples

```
library(rtiff)
tif <- readTiff(system.file("tiff", "jello.tif", package="rtiff"))
```

```
plot(tif)

plot(tif@blue)

plot(tif@blue > autoThreshold(tif@blue)[3])
```

autoThreshold	<i>Suggests threshold levels to use in binarizing an image channel.</i>
---------------	---

Description

This is an implementation of the Ridler method for binarization (see references). NOTE: The order of the values returned did not match what is described in the "value" section below. This is fixed in this version.

Usage

```
autoThreshold(d.m, est = 0.5)
```

Arguments

d.m	A data matrix representing the pixel intensities for a single image channel (e.g. <code>readTiff("image.tif")@red</code>).
est	The initial thresholding estimate to work from. The default only works if the pixel intensities are between 0 and 1, otherwise the mean intensity of the entire channel is likely a reasonable starting point. The selection of the estimate should not influence the final threshold selected.

Value

A vector (v) of estimates, the 3rd element of which is the true Ridler estimate. However, experience demonstrates that sometimes a lower (elements 1 or 2) or higher (elements 4 or 5) estimate performs better for a given application. The Ridler estimate is the mean between the average intensity of bright regions in the raster and the average intensity of dim regions. This is v[3]. v[1] is the average of dim regions, v[5] is the average of dim regions, and v[2] and v[4] are the mean between the Ridler estimate and v[1] and v[2], respectively.

Author(s)

Eric Kort <eric.kort@vai.org>

References

Ridler T, Calvard S. Picture thresholding using an iterative selection method. IEEE Trans on Systems Man and Cybernetics, SMC 8;630-2.

Examples

```
tif <- readTiff(system.file("tiff", "jello.tif", package="rtiff"))
threshold <- autoThreshold(tif@red)
plot(tif)
plot(tif@red > threshold[3])
```

getDescription	<i>Get the description on a TIFF file.</i>
----------------	--

Description

Gets the TIFF description field (tag 270) for a TIFF file.

Usage

```
getDescription(fn)
```

Arguments

fn	The name of the tiff file.
----	----------------------------

Value

A string with the description field from the TIFF file, or NA if there is no such field.

Author(s)

Eric Kort <eric.kort@vai.org>

Examples

```
# This has no description
getDescription(system.file("tiff", "jello.tif", package="rtiff"))
# but this has
getDescription(system.file("tiff", "t1.tif", package="rtiff"))
```

newPixmapRGB	<i>Create an RGB pixmap</i>
--------------	-----------------------------

Description

A utility function to create an RGB pixmap object out of 3 rasters.

Usage

```
newPixmapRGB(red = NULL, green = NULL, blue = NULL)
```

Arguments

red	The red raster (a matrix)
green	The green raster (a matrix)
blue	The blue raster (a matrix)

Value

An object of type pixmap representing the resulting RGB image.

Author(s)

Eric Kort <eric.kort@vai.org>

Examples

```
#this is tautological, but demonstrates the syntax...
tif <- readTiff(system.file("tiff", "jello.tif", package="rtiff"))
myPixmap <- newPixmapRGB(tif@red, tif@green, tif@blue)
plot(myPixmap)

#let's scramble the color channels just for fun

getOption("device")()
myPixmap <- newPixmapRGB(tif@blue, tif@red, tif@green)
plot(myPixmap)
```

`plot.matrix`*Overload the plot function for matrices*

Description

This is a simple S3 method for the plot function which takes a matrix, converts it to an RGB pixmap (by replicating it 3 times, once for each channel), and then plotting the resulting image.

Usage

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

Arguments

`x` A matrix representing an image raster.
`...` Additional arguments to send to plot.

Value

None.

Author(s)

Eric Kort <eric.kort@vai.org>

Examples

```
tif <- readTiff(system.file("tiff", "jello.tif", package="rtiff"))  
plot(tif@red)
```

`readTiff`*A function to load TIFF images into a pixmap.*

Description

Loads a TIFF image from a file and returns the image as a pixmap object, with optional scaling.

Usage

```
readTiff(fn, page = 0, reduce = 0, pixmap = TRUE)
```

Arguments

fn	Filename (the tiff image to load)
page	In the case of multi-page tiffs, which page do you want?
reduce	Optional scaling factor to improve performance with large images, should be a value between 0 and 1 (i.e. a decimal representation of a percentage). See details.
pixmap	By default, readTiff returns a pixmap object, ideal for plotting. However pixmap rescales the values to [0..1]. If you want the raw values from the file, set pixmap=FALSE to get a list of r, g, and b values, unscaled.

Details

This package is a wrapper around libtiff (www.libtiff.org), on which it depends. By using libtiff's high level TIFFReadRGBAImage function, this package inherently support a wide range of image formats and compression schemes (interestingly, thanks to libtiff, this package can load a number of TIFF formats that tools like Window's Paint or the open source Gimp application will not load).

High resolution images (by which I mean images that contain a "large" number of pixels) will occupy quite a bit of memory and will also plot very slowly. If you do not need all the resolution for your purposes, you can specify a scaling factor (`reduce=x`) to downsample the image. The factor is the amount you want the image scaled BY, not TO. In other words, `reduce=.90` will reduce the image by 90 yielding an image 10 approach is used. Since we are reducing and not enlarging, I hope this will be suitable for your applications.

For simplicity, an RGB pixmap is generated regardless of the colorspace of the original image (including grayscale images, in which case the R, G, and B rasters are identical). The pixmap object requires pixel intensities to be between 0 and 1, so the intensities in the original image file are scaled accordingly. Alternatively, specify `pixmap=FALSE` to return a list of unscaled r, g, and b values instead of a pixmap object.

Value

A pixmap object containing the image rasters.

Author(s)

Eric Kort <eric.kort@vai.org>

Examples

```
tif <- readTiff(system.file("tiff", "jello.tif", package="rtiff"))
plot(tif)
```

updateDescription	<i>Set the description on a TIFF file.</i>
-------------------	--

Description

Sets the TIFF description field (tag 270) for a TIFF file.

Usage

```
updateDescription(fn, description)
```

Arguments

fn	The name of the tiff file.
description	The text to set the description to.

Value

None.

Author(s)

Eric Kort <eric.kort@vai.org>

writeTiff	<i>A function to load TIFF images into a pixmap.</i>
-----------	--

Description

Loads a TIFF image from a file and returns the image as a pixmap object, with optional scaling.

Usage

```
writeTiff(pixmap, fn)
```

Arguments

pixmap	Either a pixmapRGB or matrix containing the image to save. In the case of a matrix, a new pixmapRGB will be created from it. The resulting TIFF file will be RGB, but will appear grey because each channel will be identical.
fn	What to call the new tiff file.

Details

This function saves the given pixmap or matrix raster as an unencrypted TIFF image, utilizing libtiff's TIFFWriteEncodedStrip, with the entire raster in a single strip (for simplicity).

Value

None. Used for its handy side effect of creating a tiff file.

Author(s)

Eric Kort <eric.kort@vai.org>

Examples

```
tif <- readTiff(system.file("tiff", "jello.tif", package="rtiff"))
writeTiff(tif@red, "atesttif.tif")
# clean up afterourselves
unlink("atesttif.tif")
```

Index

*Topic **IO**

- autoThreshold, 3
- getDescription, 4
- readTiff, 6
- rtiff-package, 2
- updateDescription, 8
- writeTiff, 8

*Topic **file**

- autoThreshold, 3
- getDescription, 4
- readTiff, 6
- updateDescription, 8
- writeTiff, 8

*Topic **utilities**

- autoThreshold, 3
- getDescription, 4
- newPixmapRGB, 5
- plot.matrix, 6
- readTiff, 6
- updateDescription, 8
- writeTiff, 8

autoThreshold, 3

getDescription, 4

newPixmapRGB, 5

plot.matrix, 6

readTiff, 6

rtiff (rtiff-package), 2

rtiff-package, 2

updateDescription, 8

writeTiff, 8