NAME

TIFFReadRGBAStrip – read and decode an image strip into a fixed-format raster

SYNOPSIS

#include <tiffio.h>

#define TIFFGetR(abgr) ((abgr) & 0xff)
#define TIFFGetG(abgr) (((abgr) >> 8) & 0xff)
#define TIFFGetB(abgr) (((abgr) >> 16) & 0xff)
#define TIFFGetA(abgr) (((abgr) >> 24) & 0xff)

int TIFFReadRGBAStrip(TIFF * *tif* **, uint32** *row*, **uint32 *** *raster*)

DESCRIPTION

TIFFReadRGBAStrip reads a single strip of a strip-based image into memory, storing the result in the user supplied RGBA *raster*. The raster is assumed to be an array of width times rowsperstrip 32-bit entries, where width is the width of the image (TIFFTAG_IMAGEWIDTH) and rowsperstrip is the maximum lines in a strip (TIFFTAG_ROWSPERSTRIP).

The *row* value should be the row of the first row in the strip (strip * rowsperstrip, zero based).

Note that the raster is assume to be organized such that the pixel at location (*x*,*y*) is *raster*[*y***width*+*x*]; with the raster origin in the *lower-left hand corner* of the strip. That is bottom to top organization. When reading a partial last strip in the file the last line of the image will begin at the beginning of the buffer.

Raster pixels are 8-bit packed red, green, blue, alpha samples. The macros *TIFFGetR*, *TIFFGetG*, *TIFFGetB*, and *TIFFGetA* should be used to access individual samples. Images without Associated Alpha matting information have a constant Alpha of 1.0 (255).

See the *TIFFRGBAImage*(3TIFF) page for more details on how various image types are converted to RGBA values.

NOTES

Samples must be either 1, 2, 4, 8, or 16 bits. Colorimetric samples/pixel must be either 1, 3, or 4 (i.e. *SamplesPerPixel* minus *ExtraSamples*).

Palette image colormaps that appear to be incorrectly written as 8-bit values are automatically scaled to 16-bits.

TIFFReadRGBAStrip is just a wrapper around the more general *TIFFRGBAImage*(3TIFF) facilities. It's main advantage over the similar *TIFFReadRGBAImage()* function is that for large images a single buffer capable of holding the whole image doesn't need to be allocated, only enough for one strip. The *TIFFReadRGBATile()* function does a similar operation for tiled images.

RETURN VALUES

1 is returned if the image was successfully read and converted. Otherwise, 0 is returned if an error was encountered.

DIAGNOSTICS

All error messages are directed to the *TIFFError*(3TIFF) routine.

Sorry, can not handle %d-bit pictures. The image had *BitsPerSample* other than 1, 2, 4, 8, or 16.

Sorry, can not handle %d-channel images. The image had *SamplesPerPixel* other than 1, 3, or 4.

Missing needed "PhotometricInterpretation" tag. The image did not have a tag that describes how to display the data.

No "PhotometricInterpretation" tag, assuming RGB. The image was missing a tag that describes how to display it, but because it has 3 or 4 samples/pixel, it is assumed to be RGB.

No "PhotometricInterpretation" tag, assuming min-is-black. The image was missing a tag that describes how to display it, but because it has 1 sample/pixel, it is assumed to be a grayscale or bilevel image.

No space for photometric conversion table. There was insufficient memory for a table used to convert image samples to 8-bit RGB.

Missing required "Colormap" tag. A Palette image did not have a required *Colormap* tag.

No space for tile buffer. There was insufficient memory to allocate an i/o buffer.

No space for strip buffer. There was insufficient memory to allocate an i/o buffer.

Can not handle format. The image has a format (combination of *BitsPerSample*, *SamplesPerPixel*, and *PhotometricInterpretation*) that *TIFFReadRGBAImage* can not handle.

No space for B&W mapping table. There was insufficient memory to allocate a table used to map grayscale data to RGB.

No space for Palette mapping table. There was insufficient memory to allocate a table used to map data to 8-bit RGB.

SEE ALSO

TIFFOpen(3TIFF), **TIFFRGBAImage**(3TIFF), **TIFFReadRGBAImage**(3TIFF), **TIFFReadRGBATile**(3TIFF), **libtiff**(3TIFF)

Libtiff library home page: **http://www.simplesystems.org/libtiff/**