MathWorks

MATLAB Answers™

CA

+ Follow

Vote 0

How can i check if an image is RGB or grayscale or binary?

Asked by Cahaya on 18 Aug 2012

Latest activity Commented on by Stephen Cobeldick MVP on 27 Jul 2018

Accepted Answer by Image Analyst MVP

257 views (last 30 days)

I'm a new in matlab, need answer to this question.. And, may i change the size of matrix of RGB image to two-dimensional matrix? I wanna try edge detection, i load an grayscale image but the image matrix show an image have three-dimensional matrix.. Any solutions?? Thank you for your response..

3 Comments

Jagdish on 24 Apr 2014

%% You can check if an image is RGB or grayscale or binary? by using imagemodel1 function.... as shown below

|Im = Imread (*xyz*.....); handle = image(Im); imgmodel = imagemodel(handle); str = getImageType(imgmodel)

Sky Dutta on 27 Jul 2018

How can i get to know that the image is rgb? I mean I need a program which tell me that "This image is Colored image or this is a binary image"?

Related Content

MATLAB Answers

My image is RGB image but when I used size function it showed me grayscale image.

2 Answers

how can i check that my image is rgb

3 Answers

Sir i am getting error :function isgray has been removed. Please suggest me which version of matlab i should use.

2 Answers

Entire Website

MATLAB image display - truecolor and indexed images

Blogs

Image Type Conversion

Documentation

Image Types

Blogs

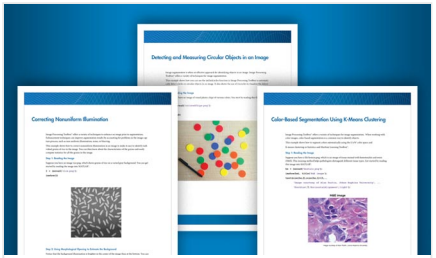
Tags

No tags entered yet.

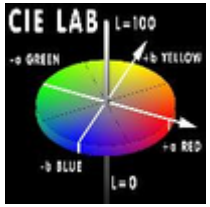
Stephen Cobeldick MVP on 27 Jul 2018

@Sky Dutta: read the comments to the accepted answer.

Sign in to comment.



3 Answers



Vote

0

Link

Answer by Image Analyst MVP on 18 Aug 2012

Accepted Answer

You can convert to grayscale using a combination of all color channels:

```
grayImage = rgb2gray(rgbImage);
```

Or you can extract one of the color channels:

```
% Extract the individual red, green, and blue color channels.
redChannel = rgbImage(:, :, 1);
greenChannel = rgbImage(:, :, 2);
blueChannel = rgbImage(:, :, 3);
```

You can get the size (dimensions) like this:

```
[rows columns numberOfColorChannels] = size(yourImageArray);
```

5 Comments

Show 2 older comments

Akib Rahman on 15 Jun 2018

Stephen Thanks for your reply. But, how can I be sure **C** is a grayscale image and **D** is a binary image? That's mean, is there any function or anything for checking image type? Thanks in advanced.

Stephen Cobeldick MVP on 15 Jun 2018

@Akib Rahman: images are just numbers, and numbers really don't tell us anything about how numbers should be interpreted. Humans attribute meaning to those numbers by their context. The image type (RGB, grayscale, binary) can be stored as meta-data (like image files do, e.g. in JPEG: "truecolor", "grayscale", etc.) or implied from the array class (but this is not always reliable, see below).

Fundamentally a binary image is just an image with only two values, so you could check if the entire image only have two values. But what happens if I used uint8 and my photo just *happens* to have exactly two colors in it?

Consider an image that only contains black pixels: there is no way to know *from the values alone* if the image is intended to be binary or grayscale. The data type could be used (e.g. a logical / one-bit-per-pixel-class can only be used to encode a binary image), but consider if I use uint8 to encode a binary image and then send it to you: in this case you have no way to know if my image is binary or grayscale or even indexed. These would all be exactly the same size. There is no way to "know" what the image type is, unless I tell you or give you supplementary information (e.g. a map, the bit level per pixel, etc).

Walter Roberson MVP on 15 Jun 2018

```
ndims(A) ==2 && islogical(A) --> true if binary image
ndims(A) == 2 && length(unique(A(:))) == 1 --> true if grayscal
ndims(A) == 2 && length(unique(A(:))) == 2 --> true if grayscal
ndims(A) > 2 --> cannot be grayscale or indexed or binary
ndims(A) == 3 && size(A,3) == 3 && size(unique(A, 'rows'),1) ==
ndims(A) == 3 size(A,3) == 3 && size(unique(A, 'rows'),1) == 2
```

Here, bi-level refers to images that have two distinct colors. They might be binary images, but they might not be (for example, red letters on blue background). It is common to encode binary images as the values 0 and 1, but it is also common to encode binary images as the values 0 and 255.

Sign in to comment.

Vote

0

Link

Answer by Walter Roberson MVP on 18 Aug 2012

Use rgb2gray() to convert the image to grayscale.

0 Comments

Sign in to comment.

Vote

0

Link

Answer by Noor Ul Islam on 27 Dec 2013

% first Read image

```
img=imread('imag.jpg');
```

% Check img whether it rgb or grayscale

```
[r c d]=size(img)
% if image is 3D above command will give an error; solution=remove o/p v
```

% to convert to grayscale

```
% use inbuilt command
```

```
img1=rgb2gray(img);  %now you get 2D image
```

```
% to manipulate the true colors RGB; split image into R,G,B compone
```

```
        R=img(:,:,1);
        G=img(:,:,2);
        B=img(:,:,3);
%Note: each R,G and B component is 2D matrix... hope this is what u requ
```

0 Comments

Sign in to comment.

Sign in to answer this question.