bit.ly/clps950_2pm

Images

today we move from 2d to 3d!

10 minute warm up

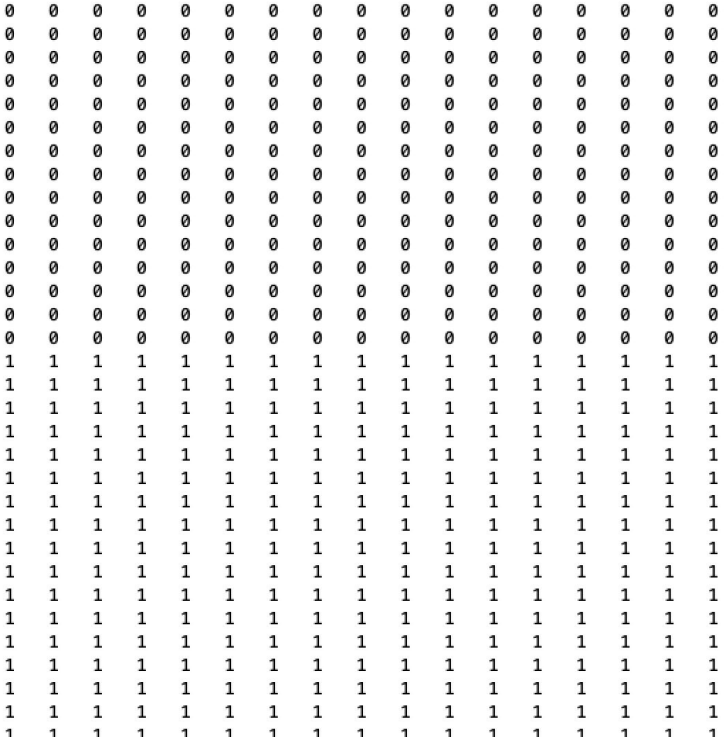
1. create a 5x5 matrix z which has 8 on the diagonal and -2 everywhere else
2. set n to the sum of **all** the elements in the even numbered rows (n is single number)
3. set m to a vector of the max value in each row of z
4. make a new matrix z2 which has m*3 as its first column, followed by z for other columns

10 minute warm up

1. create a 5x5 matrix z which has 8 on the diagonal and -2 everywhere else
2. set n to the sum of **all** the elements in the even numbered rows (n is single number)
3. set m to a vector of the max value in each row of z
4. make a new matrix z2 which has m*3 as its first column, followed by z for other columns

```
% create a 5x5 matrix z which has 8 on the diagonal and -2 everywhere else
z = eye(5)*10 - 2;
% set n to the sum of the elements in the even numbered rows
n = sum(z(2:2:end,:),'all');
% set m to a vector of the max value in each row of z
m = max(z,[],2);
% make a new matrix z2 which has m*3 as its first column, followed by z
z2 = [m*3 z];
```

What is an image?



It is an array, with dimensions

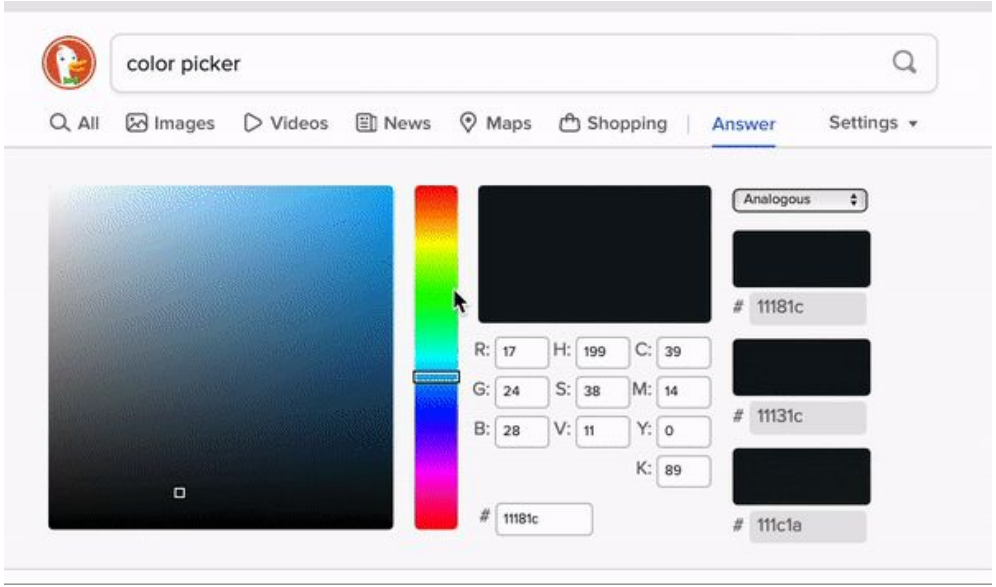
HxWx1 if its a grayscale image

HxWx3 if its a color image.

Color image, each entry is a triplet (R,G,B). Which is the color of each pixel.

All colors can be represented with 3 numbers!

RGB is one encoding scheme, if you're more familiar with hex codes. Try typing 'color picker' into your search engine to try out some colors



What is an image? (Lets see what is inside this matrix)

190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
191	191	191	191	191	191	191	191	191	191	191
190	190	190	190	190	190	190	190	190	190	190
190	190	190	190	190	190	190	190	190	190	190
253	253	253	253	253	253	253	253	253	253	253
252	252	252	252	252	252	252	252	252	252	252
251	251	251	251	251	251	251	251	251	251	251
252	252	252	252	252	252	252	252	252	252	252
252	252	252	252	252	252	252	252	252	252	252
252	252	252	252	252	252	252	252	252	252	252
252	252	252	252	252	252	252	252	252	252	252

img = webread('https://tinyurl.com/clps0950ex');

Can anyone tell what is the image??

To visualize an image use imgesc or imshow.

imgesc(img)

imshow(img)

RGB vs Grayscale (rgb2gray)

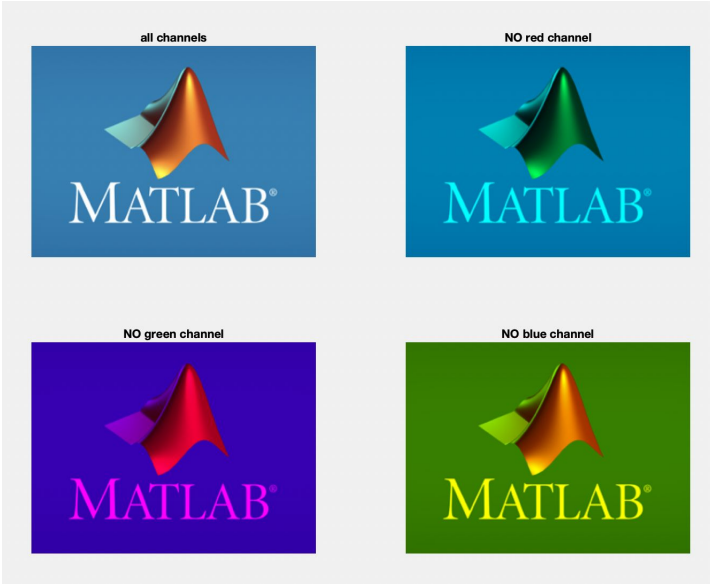
Grayscale only has one channel. Remember (HxWx1) , which means that it only captures values of brightness being 0 lowest (just dark) and 255 highest (white).

You can convert easily a color image into a grayscale. (but not the other way)

subplot(1,2,1), imshow(img); subplot(1,2,2), imshow(rgb2gray(img))

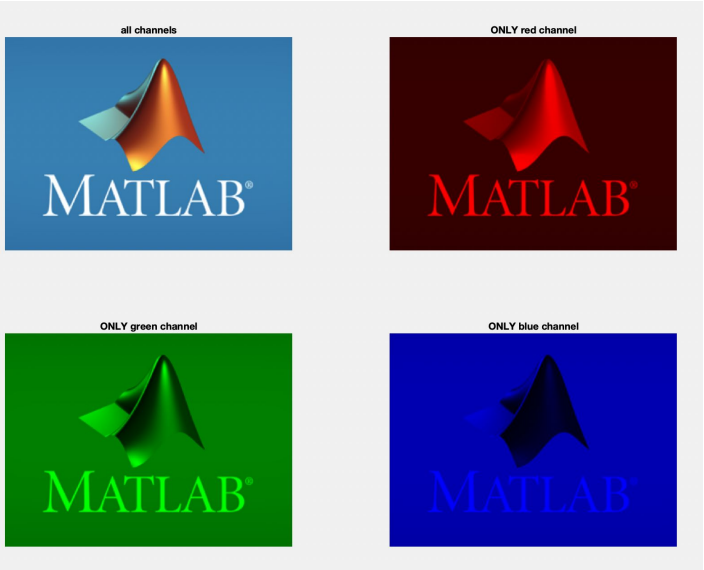
What does it look like to decompose into channels?

```
img = webread('https://haleyk.github.io/matlab_logo.png');
subplot(2,2,1)
imshow(img); title('all channels')
subplot(2,2,2)
imnored = img; imnored(:,:,1) = 0;
imshow(imnored); title('NO red channel')
subplot(2,2,3)
imnogreen = img; imnogreen(:,:,2) = 0;
imshow(imnogreen); title('NO green channel')
subplot(2,2,4)
imnoblue = img; imnoblue(:,:,3) = 0;
imshow(imnoblue); title('NO blue channel')
```



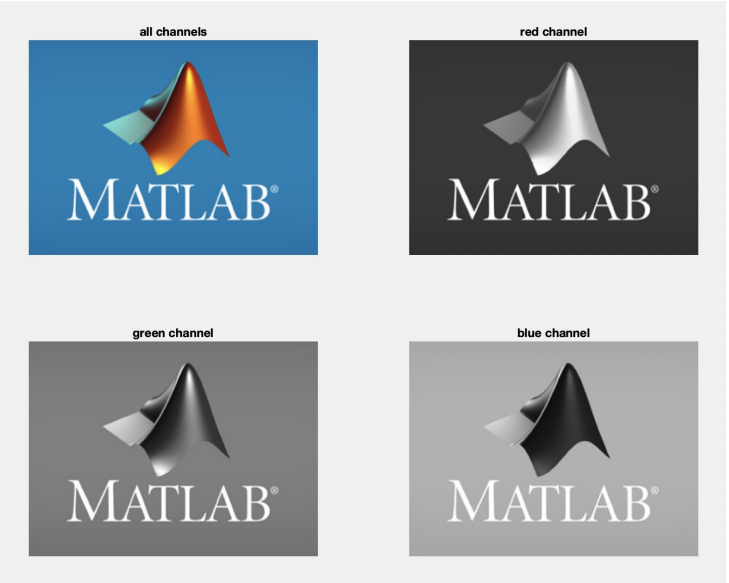
What does it look like to decompose into channels?

```
img = webread('https://haleyk.github.io/matlab_logo.png');
subplot(2,2,1)
imshow(img); title('all channels')
subplot(2,2,2)
imred = img; imred(:,:,2) = 0; imred(:,:,3) = 0;
imshow(imred); title('ONLY red channel')
subplot(2,2,3)
imgreen = img; imgreen(:,:,1) = 0; imgreen(:,:,3) = 0;
imshow(imgreen); title('ONLY green channel')
subplot(2,2,4)
imblue = img; imblue(:,:,1) = 0; imblue(:,:,2) = 0;
imshow(imblue); title('ONLY blue channel')
```



What does it look like to decompose into channels?

```
img = webread('https://haleyk.github.io/matlab_logo.png');
subplot(2,2,1)
imshow(img,[0 255]); title('all channels')
subplot(2,2,2)
imshow(img(:,:,1),[0 255]); title('red channel')
subplot(2,2,3)
imshow(img(:,:,2),[0 255]); title('green channel')
subplot(2,2,4)
imshow(img(:,:,3),[0 255]); title('blue channel')
```



think: why are they black and white?

Practice – Cropping Images and Removing Information

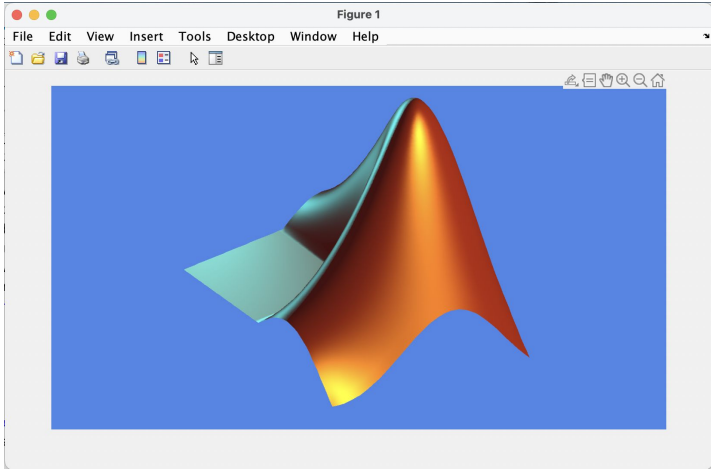
How would we crop out the word of that matlab image?

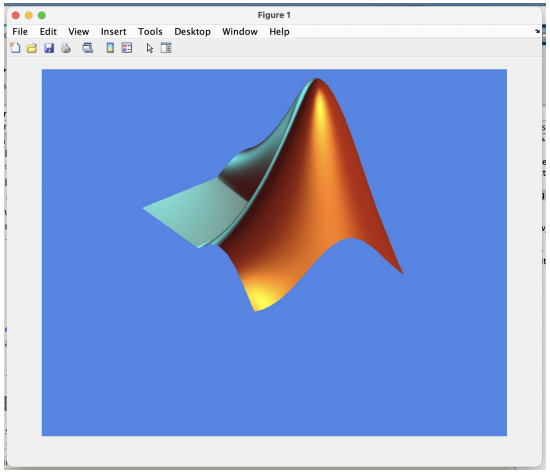
How could we (start) to remove the WHITE parts of the matlab image (the word matlab)? (to remove, I mean set to the background)

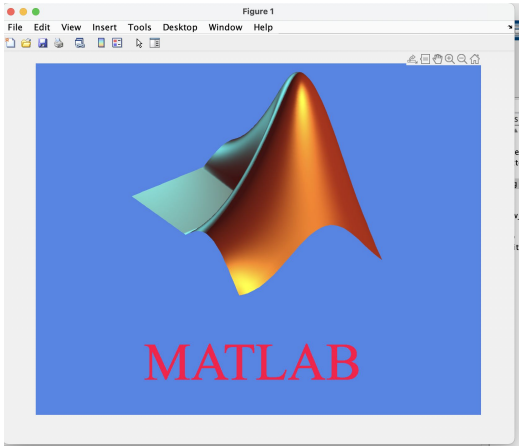
Instead of removing it, what if we want to turn it red?

hint! you will have to use loops!

```
blue = [88 132 225];
white = [255 255 255];
```







yourimage = webread('https://haleyk.github.io/matlab_basic.png');
imshow(yourimage)

Solutions

```
img = webread('https://haleyk.github.io/matlab_basic.png');

blue = [88 132 225];
red = [239 38 75];
white = [255 255 255];

% problem 1: cropping out the bottom
top = img(1:799,:,:); % what I want to keep
bottom = img(800:end,:,:); % what I don't need
figure; imshow(top)

% problem 2: making the bottom blue
% can be done this way OR same as #3, replacing red with blue
bottom(:,:,1) = 88; % red channel value
bottom(:,:,2) = 132; % blue channel value
bottom(:,:,3) = 225; % green channel value
% or if you like loops...
% for channel=1:3
%     bottom(:,:,channel) = blue(channel);
% end
new_image = [top; bottom];
figure; imshow(new_image)
```

```
% problem 3: making the text red
new_bottom = img(800:end,:,:); % fresh copy
new_b = NaN(size(new_bottom)); %create empty version
for row=1:size(new_bottom,1) % loop through rows
    for col=1:size(new_bottom,2) % loop through columns
        % squeese gets rid of the 1s at the start
        pixel_color = squeeze(new_bottom(row,col,:));
        if (pixel_color(1)==white(1)) && (pixel_color(2)==white(2)) && (pixel_color(3)==white(3))
            new_b(row,col,:) = red; % set all 3 channels to the 3 values in my red variable
        elseif (pixel_color(1)==blue(1)) && (pixel_color(2)==blue(2)) && (pixel_color(3)==blue(3))
            new_b(row,col,:) = blue; % set all 3 channels to the 3 values in my blue variable
        else % not white or blue... in this case, let's just do blue.
            new_b(row,col,:) = blue;
        end
    end
end
red_text_image = [top; new_b]; % put them back together
figure; imshow(red_text_image)
```