

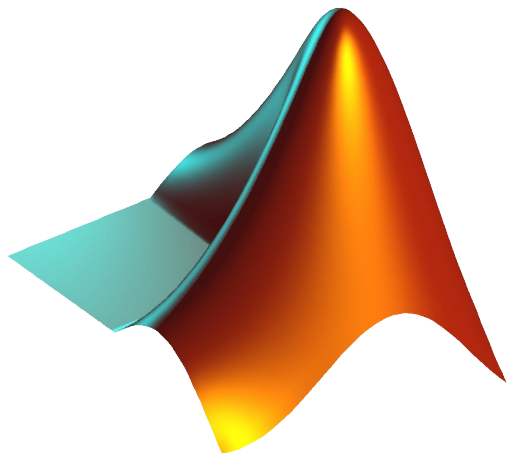
# CS 1112 Introduction to Computing Using MATLAB

Instructor: Dominic Diaz

Website:

<https://www.cs.cornell.edu/courses/cs1112/2022fa/>

Today: images in color

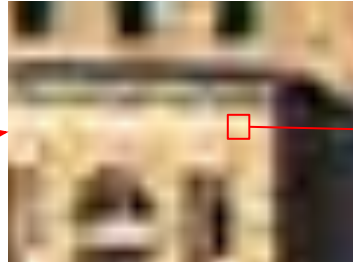


# Agenda and announcements

- Last time
  - Image processing
    - Add frame on grayscale image (2D array)
    - Imread and imshow
- Today
  - Images in color (3D array)
- Announcements
  - Project 4 released (due 10/26)
  - Partner service out! Submit if you need a partner!
  - Prelim 1 is today from 7:30 - 9 in Klarman Hall (KG70). Students with approved exceptions please check CMS for your time/location.

Any prelim questions?

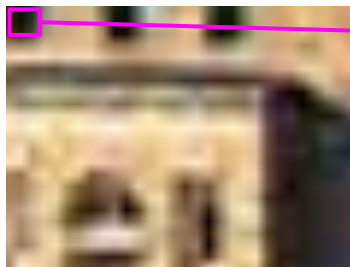
# Images are made up of pixels



Each pixel's color is defined by a 3 uint8 numbers

152	95	133
-----	----	-----

# Color images are represented by 3D arrays



				5	6	27	16	76	5	
				6	4	99	65	89	2	23
			10	5	200	198	189	5	16	75
			9	3	68	61	56	4	189	52
			117	126	183	135	186	199	132	2
			99	105	105	109	112	124	115	
			89	88	95	98	100	105		

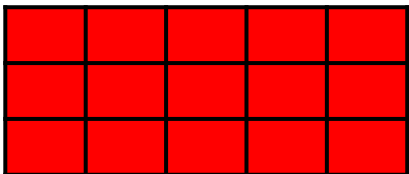
For a uint8 array, all elements in the array are integers between 0 and 255, inclusive.

- Values closer to [0, 0, 0] are closer to black
- Values closer to [255, 255, 255] are closer to white

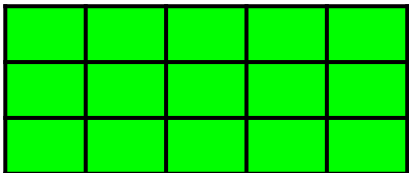
# 3D array indices

```
[nr, nc, np] = size(A) % stores the dimensions of a 3D array A  
% nr stores number of rows  
% nc stores number of columns  
% np stores number of panes (for color images, np = 3)
```

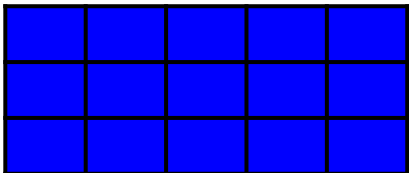
```
A(1:nr, 1:nc, 1)
```



```
M1 = A(:, :, 1); % red values for each pixel
```



```
M2 = A(:, :, 2); % green values for each pixel
```



```
M3 = A(:, :, 3); % blue values for each pixel
```

# Example: mirror image



Cornell University Law School  
Photograph by Cornell University Photography



Cornell University Law School  
Photograph by Cornell University Photography

Flip image horizontally

1. Read image from memory and convert it into an array
2. Manipulate the array
3. Convert the array to a jpg file and write it to memory

## Reading and writing image files

```
% Read image file, store in a 3D array A of  
% type uint8.
```

```
A = imread('LawSchool.jpg');
```

```
% Write 3D array B to memory as jpg image  
imwrite(B, 'LawSchoolMirrored.jpg')
```



File name for the image you are storing



```
A = imread('LawSchool.jpg');  
[nr, nc, np] = size(A);
```

```
B = zeros(size(A));
```

```
% loop through B and set value in B to the appropriate  
% values in A
```

```
imwrite(B, 'LawSchoolMirrored.jpg')
```

```
A = imread('LawSchool.jpg');
[nr, nc, np] = size(A);

B = zeros(size(A));

for r = 1:nr
    for c = 1:nc

        for p = 1:np
            % set B(r,c,p) to appropriate value in A
        end

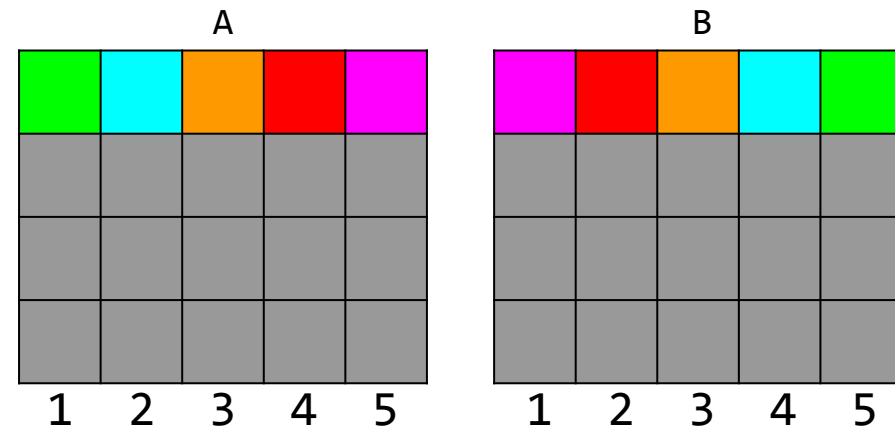
    end

end
imwrite(B, 'LawSchoolMirrored.jpg')
```

```
A = imread('LawSchool1.jpg');  
[nr, nc, np] = size(A);
```

```
B = zeros(size(A));
```

```
for r = 1:nr  
    for c = 1:nc  
        for p = 1:np  
            B(r,c,p) = A(r, nc-c+1,p);  
        end  
    end  
end  
imwrite(B, 'LawSchoolMirrored.jpg')
```



Column c in B  
is column  $nc-c+1$  in A

```
A = imread('LawSchool.jpg');
[nr, nc, np] = size(A);

B = zeros(size(A)); % B a 3D array of doubles
B = uint8(B);      % transform B to 3D array of uint8s

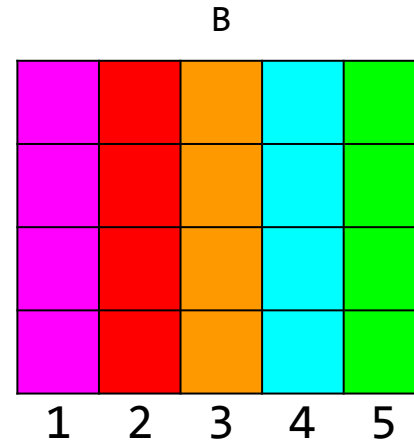
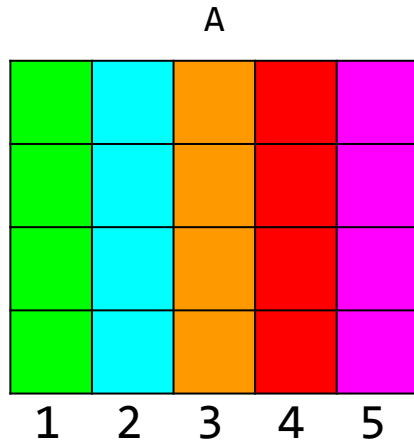
for r = 1:nr
    for c = 1:nc

        for p = 1:np
            B(r,c,p) = A(r, nc-c+1,p);
        end

    end

end
imwrite(B, 'LawSchoolMirrored.jpg')
```

Vectorized code simplifies things... work with a whole column at a time



Column  $c$  in B

is column  $nc-c+1$  in A

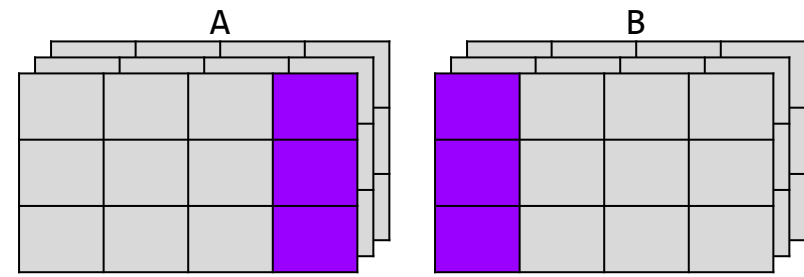
```
A = imread('LawSchool.jpg');  
[nr, nc, np] = size(A);
```

```
B = zeros(size(A));  
B = uint8(B);
```

```
for c = 1:nc
```

```
    B(:,c,1) = A(:,nc-c+1,1); % Copy over red values
```

```
end  
imwrite(B, 'LawSchoolMirrored.jpg')
```



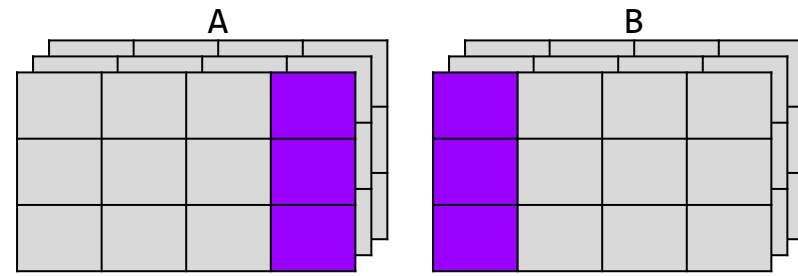
```
A = imread('LawSchool.jpg');  
[nr, nc, np] = size(A);
```

```
B = zeros(size(A));  
B = uint8(B);
```

```
for c = 1:nc
```

```
    B(:,c,1) = A(:,nc-c+1,1); % Copy over red values
```

```
end  
imwrite(B, 'LawSchoolMirrored.jpg')
```



When  $c = 1$

```
A = imread('LawSchool.jpg');  
[nr, nc, nl] = size(A);
```

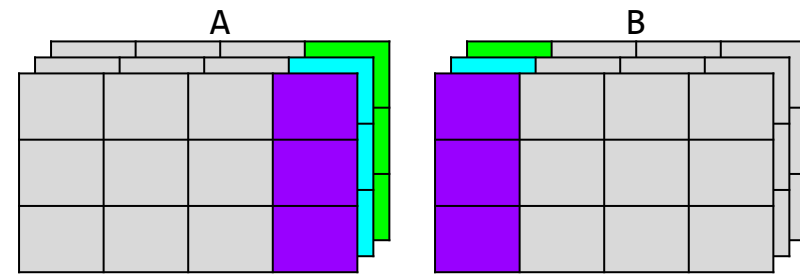
```
B = zeros(size(A));  
B = uint8(B);
```

```
for c = 1:nc
```

```
    B(:,c,1) = A(:,nc-c+1,1); % Copy over red values  
    B(:,c,2) = A(:,nc-c+1,2); % Copy over blue values  
    B(:,c,3) = A(:,nc-c+1,3); % Copy over green values
```

```
end
```

```
imwrite(B, 'LawSchoolMirrored.jpg')
```



When c = 1



```
A = imread('LawSchool.jpg');
[nr, nc, nl] = size(A);

B = zeros(size(A)); % B a 3D array of doubles
B = uint8(B);      % transform B to 3D array of uint8s

for c = 1:nc

    B(:,c,:) = A(:,nc-c+1,:); % Copy over all colors in
                                % in a single column

end
imwrite(B, 'LawSchoolMirrored.jpg')
```