Image Filtering

Examples
RGB Color images

• Color images are typically stored as red, green, blue

```matlab
>> RGB = imread('peppers.png');
>> imshow(RGB)
>> whos
Name         Size         Bytes  Class  Attributes
RGB  384x512x3       589824  uint8
```

• To convert to grayscale, can use
  – G = rgb2gray(RGB);
Indexed color images

- Indexed color images have an associated colormap.
- Each pixel is an index into the map
- \([I, cmap] = \text{imread('forest.tif')}\);
- To convert to nonindexed, do
- \(\text{RGB} = \text{ind2rgb}(I, cmap)\);
Enhancing low contrast images

• Simple scaling is often enough
  \[ \text{imshow}(I, []); \quad \text{scales as it displays (but doesn’t change I)} \]
  \[ I2 = \text{imadjust}(I1); \quad \text{scales and outputs the scaled image} \]

• If the image has a complicated histogram, then use histogram equalization
  \[ - I2 = \text{histeq}(I); \]
  \[ - I2 = \text{adapthisteq}(I); \quad \text{Adaptive histogram equalization} \]

• Try on ‘forest.tif’”
• Find a low contrast image and enhance it
Gradient Magnitude and Direction

• Read the “coins.png” image. Compute the gradient of the image by filtering the image with the Sobel operators in the horizontal and vertical directions. Display the gradient magnitude and the gradient angles.

• Threshold the gradient magnitude image, using an expression such as “T = GM > v;”, where GM is the gradient magnitude image and v is some value that you select experimentally. The resulting threshold image T should have white (1’s) where the gradient magnitude was larger than v, and black (0’s) where the gradient magnitude was less than or equal to v.

• Multiply the threshold image with the gradient angle image. This should zero out the noisy angle values, corresponding to the places where the gradient magnitude was too small. Display the resulting gradient angle image.