ECE 3552 Lab 3: Image Processing in MATLAB part 2

Digital Image processing using the toolbox:

Digital image processing is the use of computer algorithms to create, process, communicate, and display digital images. Digital image processing algorithms can be used to:

Convert signals from an image sensor into digital images

Improve clarity, and remove noise and other artifacts

Extract the size, scale, or number of objects in a scene

Prepare images for display or printing

Compress images for communication across a network

Image Enhancements:

MATLAB allows a variety of image enhancements, which can be applied ideally before further analysis to identify interesting regions or details of an image. Some basic enhancements are as follows:

Deblurring

Filtering

Morphological Operations

Image arithmetic

Colorspace transforms

Contrast adjustments

We have seen a few of these methods in the previous labs. A more complete list of operations can be found here: https://www.mathworks.com/help/images/functionlist.html

Segmentation:

Image segmentation is the process of dividing an image into multiple parts. This is typically used to identify objects or other relevant information in digital images. There are many different ways to perform image segmentation, including:

Thresholding

Color Based

Watershed methods

Texture filtering

Following are brief tutorials for each of the above mentioned methods:

Thresholding:

https://www.mathworks.com/help/images/correcting-nonuniform-illumination.html

Color Based:

https://www.mathworks.com/help/images/color-based-segmentation-using-k-means-clustering.html?prodcode=IP&language=en

Watershed segmentation:

https://www.mathworks.com/help/images/marker-controlled-watershed-segmentation.html?prodcode=IP&language=en

Texture Filter:

https://www.mathworks.com/help/images/texture-segmentation-using-texture-filters.html

Raspberry PI image acquisition instructions:

https://www.raspberrypi.org/documentation/configuration/camera.md

https://www.raspberrypi.org/documentation/raspbian/applications/camera.md

Exercise:

1: Go through the above linked tutorials, and display the results of each segmentation method.

2: Apply a simple deblur/denoise (for example, deblur with Weiner filter) to the test images provided, and then perform the segmentation operations discussed previously. Which operation is more robust to blurry images? Noisy images?

3: Repeat step 3 with a different deblur/denoise operation and compare your results. Repeat with images acquired from your Raspberry Pi Camera.