EN2400 – Image Processing

Image Processing in MATLAB - A Short Tutorial

March 23, 2009

1 Image Representation

Images are typically represented using matrices in MATLAB. Gray scale images are stored in two dimensional matrices, X, where each element corresponds to the gray level of a pixel in the image. The element X(1,1) represent the top left pixel in the image, and X(Nr,Nc) is the bottom right (Nr - number of rows, Nc - number of columns). Color images are most often represented in the RGB color space using three dimensional matrices. The elements in X(:,:,1) are the red components, X(:,:,2) are the green components, and X(:,:,3) are the blue components.

2 Reading and Writing Images from/to Disk

To import and export images in MATLAB is particularly easy using the imread and imwrite commands. For example, to read a JPEG file into a matrix X, the syntax is:

```
>> X=imread('filename.jpg');
```

If the file contains a gray scale image, X is a two-dimensional matrix, and if the file contains a color image, the matrix is 3−, or 4−dimensional, depending on the color space. Note that imread handles most commonly used image formats such as BMP, PNG, GIF, and TIFF. Use

```
>> help imread
```

for more info. imread also supports reading of files using the HTTP protocol:

```
>> X=imread('http://url to image');
```

Similarly, to write an image contained in a matrix X to 'filename.jpg',

```
>> imwrite(X,'filename.jpg');
```

3 Data Types

In most of the image file formats supported by imread, pixels are stored using 8 or fewer bits per color plane. If the file contains only 1 bit per pixel, the class of the

output is logical. When reading other files with 8 or fewer bits per color plane, the class of the output is *uint8*. `imread` also supports reading 16-bit-per-pixel data from BMP, PNG, JPEG, and TIFF files. Many of `MATLAB`'s more specialized functions are defined only for the double data type. Conversion from, e.g., *uint8* is possible with the `double` command:

```
>> X=double(X);
```

4 Displaying Images

An image, `X`, can be displayed using the `image` or `imagesc` commands.
`>> imagesc(X)` `imagesc` displays matrix `X` as an image. Each element of `X` specifies the color of a sub pixel in the image. Depending on the data type, and dimensionality of the matrix, color information is interpreted in different ways. In particular, if the matrix is two-dimensional, the elements of `X` are used as indices into the current `colormap` to determine the color. If you are trying to view an eight bit gray scale image

```
>> colormap gray(256);
```

is probably what you are looking for. `imagesc` is the same as `image`, but data is scaled to use the full colormap (note that you can specify the scaling using arguments). Another useful command is to apply

```
>> axis equal
```

which will preserve the height/width ratio of the image. See the documentation for `image`, `imagesc`, `colormap`, and `axis` for more information.

5 Color to Gray Scale Transformation

When converting a color RGB image into gray scale, it is appropriate to let the gray scale represent the luminance L of the image. It is often computed as

$$L = 0.299R + 0.587G + 0.114B$$

or, according to ITU-R Rec. BT.709,

$$L = 0.2126R + 0.7152G + 0.0722B.$$

The non-uniform weighting takes into account that the human visual system has varying sensitivity to the different RGB color components.

6 Image Processing Tools in MATLAB

A plethora of signal processing tools suitable for image processing are readily available in `MATLAB`. Here, we only give a few pointers.see the information contained in `MATLAB`'s

help for more information. Some MATLAB commands of interest are: `filter`, `filter2`, `fft`, `fft2`, `fftshift`, `dct`, `conv`, `conv2`.