COMP 478/6771 (FALL 2020)

Digital Image Processing

Introduction to
Image Processing Toolbox

Instructor: Prof. Yiming Xiao

Tutors:

Materials provided by Dr. T. D. Bui

UNIVERSITÉ
Concordia
UNIVERSITY

Image Processing Toolbox

- Supports a wide range of image processing operations:
  - Spatial / Frequency transformation
  - Filtering
  - Debluring
  - Denoising
  - …

- To get a list of functions available:

  >> `help images`

`help images`

- Image display and exploration.
- Image file I/O.
- Image arithmetic.
- Spatial transformations.
- Pixel values and statistics.
- Image enhancement.
- …

Image Processing Toolbox (cont.)

- Lastest version: 6.4 (Release 2009b)
  - To check the current version, type:

    ```
    >> ver
    ```

- Not all image formats are supported!
  - To see what your installation supports:

    ```
    >> imformats
    ```

- Whos
  - Information about your variables

imformats

```
>> imformats

EXT        ISA      INFO       READ      WRITE      ALPHA   DESCRIPTION
-------------------------------------------------------------------------------------
bmp        isbmp    imbmpinfo  readbmp   writebmp   0       Windows Bitmap (BMP)
cur        iscur    imcurinfo  readcur              1       Windows Cursor resources (CUR)
fts fits   isfits   imfitsinfo readfits             0       Flexible Image Transport System (FITS)
gif        isgif    imgifinfo  readgif   writegif   0       Graphics Interchange Format (GIF)
hdf        ishdf    imhdfinfo  readhdf   writehdf   0       Hierarchical Data Format (HDF)
ico        isico    imicoinfo  readico              1       Windows Icon resources (ICO)
j2c j2k    isjp2    imjp2info  readjp2   writej2c   0       JPEG 2000 (raw codestream)
jp2        isjp2    imjp2info  readjp2   writejp2   0       JPEG 2000 (Part 1)
jpf jpx    isjp2    imjp2info  readjp2              0       JPEG 2000 (Part 2)
jpg jpeg   isjpg    imjpginfo  readjpg   writejpg   0       Joint Photographic Experts Group (JPEG)
pbm        ispbm    impnminfo  readpnm   writepnm   0       Portable Bitmap (PBM)
pcx        ispcx    impcxinfo  readpcx   writepcx   0       Windows Paintbrush (PCX)
pgm        ispgm    impnminfo  readpnm   writepnm   0       Portable Graymap (PGM)
png        ispng    impnginfo  readpng   writepng   1       Portable Network Graphics (PNG)
pnm        ispnm    impnminfo  readpnm   writepnm   0       Portable Any Map (PNM)
ppm        isppm    impnminfo  readpnm   writepnm   0       Portable Pixmap (PPM)
ras        isras    imrasinfo  readras   writeras   1       Sun Raster (RAS)
tif tiff   istif    imtifinfo  readtif   writetif   0       Tagged Image File Format (TIFF)
xwd        isxwd    imxwdinfo  readxwd   writexwd   0       X Window Dump (XWD)
```

Reading, Writing and Displaying Images

- Basic image I/O functions:
  - `imread`
  - `imwrite`
  - `imshow`

Reading, Writing and Displaying Images

- Examples:

```
>> I = imread('sample.bmp');   % Reads image
   %'sample.bmp'which is located in the working
   % directory.
>> display(I);   % Displays I
```

- Note:
  - If the image is not in the working directory, you can add the complete path.
    - You may also find these command useful: `pwd`, `cd`.

Reading, Writing and Displaying Images

- `I` is a matrix
  - 2D if it represents a binary/grayscale image
  - 3D if it represents a colour image

- Useful functions:
  - ndims   : To check the number of dimensions
  - rgb2gray   : To convert a colour image to grayscale

Reading, Writing and Displaying Images

- If `I` is a grayscale image, you can extract the top half of the image using:

  ```
  >> r = size(I, 1);
  >> c = size(I, 2);
  >> I_tophalf = I(1:floor(r/2), 1:c);
  >> % or even make it shorter: I(1:floor(r/2), :)
  ```

  - Exercise: Make it work for colour images as well.

- Write the result to a new file:

  ```
  >> imwrite(I_tophalf, 'sample2.bmp');
  ```

Representation of Images in MATLAB

- **Binary:**
  A binary image with M rows and N columns is represented by an M×N logical matrix where pixel values are 1 (true) or 0 (false).

- **Grayscale**:
  A grayscale image with M rows and N columns is represented by a M×N matrix of type:
  - `double`
  - `uint8`

- **Colour:**
  - **True-color RGB:**
    A RGB image is represented by a three-dimensional M×N×3 double matrix.
    Each pixel has red, green, blue components along the third dimension with values in [0,1]. E.g., the color components of pixel (m,n) are I(m,n,1) = red, I(m,n,2) = green, I(m,n,3) = blue.
  - **Indexed:**
    Indexed (paletted) images are represented with an index matrix of size M×N and a colormap matrix of size K×3.

Reading, Writing and Displaying Images

- Warning!
  - Always check the type and number of dimensions of an image before doing any processing:
    ```
    >> class(I)
    >> ndims(I)
    ```
  - If you need more precision, i.e. you need to convert a uint8 gray image to a double:
    ```
    >> I2 = double(I);
    >> I2 = I2 / 255; % scale it to [0, 1]
    ```
    - You could simply call the function **im2double.**
    - Another useful function is **minmax.**

Manipulating Images

- The same way as we manipulate matrices!
  >> I(1,1)
  >> I(1,1) = max(255, I(1,1) + 25)

Example 1

- close all
- I = imread('pout.tif');
- imshow(I)
- whos

Example 2

- figure,
- imhist(I)
- I2 = histeq(I);
- figure
- imshow(I2)
- imwrite (I2, 'pout2.png');
- imfinfo('pout2.png')

Example 3

- Read indexed image
  - [X,map] = imread('trees.tif');

- Get image from imshow
  - imshow('moon.tif');
  - Moon = getimage;
- Setting initial magnification
  - imshow('pout.tif', 'InitialMagnification', 150)

Example 4

- Displaying in Separate Figures
  - imshow(I)
  - figure, imshow(J)
  - figure, imshow(K)

Example 4

- **Multiple Display**
  - [X1,map1]=imread('forest.tif'); [X2,map2]=imread('trees.tif');
  - subplot(1,2,1), subimage(X1,map1) subplot(1,2,2), subimage(X2,map2)

- Showing binary images (BW = im2bw(X1, 0.5) in different color
  - imshow(BW,[1 0 0; 0 0 1]) or
  - imtool(BW,[1 0 0; 0 0 1])

Basic Image Operations with DIP Toolbox

- Some useful functions for now:
  - **imadd**
  - **Imsubtract**
  - **imcomplement**
  - ...
  - **imhist**
  - **histeq**
  - **Imadjust**
  - ...

- Exercise: Write a function to flip an image vertically.

Further Reading

- **Digital Image Processing Using MATLAB**

  R. C. Gonzelaz, R. E. Woods and S. L. Eddins

- **Image Processing Toolbox User Guide**

  http://www.mathworks.com/access/helpdesk/help/pdf_doc/images/images_tb.pdf