Background

- Command Window

- Workspace = set of variables

- Workspace Browser = shows variables

- Array Editor = double click on variable for more info

- current directory

- search path for m files; set path from file menu

- Command History Windows = current and previous sessions

Background

- MATLAB editor = edit in command window

- HELP - click on ? on desktop toolbar
  - type "help browser"

  - navigator pane

  - display pane – e.g. search tab

- doc – e.g. doc format shows format command

Images

- picture is given as a matrix

- (0,0) or (1,1) is upper left corner

- (0,1) is top row second column second index is rows (C)

- $f(x,y)$ is intensity at $(x,y)=(r,c)$

- types: TIFF,JPEG,GIF,BMP,PNG,XWD

- PNG: Portable Network Graphics; .png

- XWD: X Window Dump; .xwd

Simple Commands

- read image: f=imread('chestxray.jpg');

- `f=imread ('d:\ myimages\ chestxray . j pg');`

- [M, N] = size(f);

- whos;

- imshow(f);

- use impixel to get intensity of individual pixel - use with mouse

- MATLAB replaces next picture in same spot.
  To get a new graph use "figure"

- imshow(f), figure, imshow(g)

- imwrite(f, 'patient','tif'); or else imwrite(f,'patient.tif');

- options to change intensity range and compression

- imfinfo bubbles25.jpg or K=imfinfo('bubbles25.jpg');

Data Classes

- double = double precision; 8 bytes per element

- uint8 = unsigned 8 bit integer [0,255] byte per element

- uint16= unsigned 8 bit integer [0,65535] 2 bytes per element

- uint32= unsigned 8 bit integer [0,65535] 4 bytes per element

- int8 = signed 8 bit integer [-128,127]

- int16

- int32

- single = single precision; 4 bytes per element

- char = Characters ; 2 bytes per element

- logical= 0 or 1; 1 byte per element

Image Types

- intensity images

- binary images

- indexed images

- RGB images

conversion between classes

- im2uint8 : to uint8

- im2uint16: to uint16

- mat2gray: from double to double in [0,1]

- im2double: to double

- im2bw : to logical (binary)

Array indexing

- v = [1 3 5 7 9]; row vector

- w= v.';   column vector

- v(1:3)   -----→   1 3 5

- v(3:end)   -----→   5 7 9

- v(:) column vector

- v(1:end) row vector

- v(1:2:end) all odd components 1 5 9

- v(end:-2:1) all odd components backwards 9 5 1

- x = linspace(a, b, n);

- reshape changes size of matrix
In general, RESHAPE(X,SIZ) returns an N-D array
with the same elements as X but reshaped to the size
SIZ. PROD(SIZ) must be the same as
PROD(SIZE(X)).

Matrix indexing

- A = [1 2 3; 4 5 6; 7 8 9]; 3 x 3 matrix

- A(2, 3)   ----->   6

- C3 = A(:,3) yields 3rd column (column vector)

- R2 = A(2, :) yields 2nd row (row vector)

- s= sum(A); column vector with sum of each row

- s= sum(A(:)); or sum(sum(A)) sum of entire matrix

Applications to Images

- fp = f(end:-1:1, :); flips image vertically (1 st index is columns)

- fc = f(257:768, 267:768); crops picture

- fs = f(1:2:end, 1:2,end) subsample

standard matrices

- zeros(M,N) M columns N rows

- ones (M, N)

- true (M,N)

- false(M, N)

- rand (M,N) uniformly distributed in [0 1]

- randn(M,N) normally (Gaussian) distributed
  - mean 0 variance 1

m files

- filename.m created with text editor

- H1 line

- help text

- function body

- comments

- function [s, p] = sumprod (f,g)
  where f,g are two images and s and p are their sum and product

- H1 is first text line. It is a single comment line

- help sumprod H 1 line
  for example: SUMPROD computes the sum and product of two images

- m files can be created with any text editor

- edit sumprod opens meditor editor

Arithmetic Operators

- A*B matrix multiplication

- A.*B element by element multiplication

- imadd adds 2 images

- imsubtract, immultiply,imdivide (element by element)

Relational Operators

- A==B gives 1 where the elements are the same and 0 otherwise

    > = gives 1 where the elementof A is larger than B otherwise 0

- A&B gives 1 where the both elements are zero and 0 otherwise

- pi and i=sqrt(-1) are pre-defined

Flow Control

- if expression
  statements
  end

- for index=start:increment:end

- while expression
  statements
  end

- break

- switch expression
  case
  end

Loops

Example:
```
 if (isinf (x) || ˜isreal (x)
    disp ('Bad input')
    y = Nan;
 elseif (x==round (x) ) && (x>0)
    y = prod (1 : x-1) ;
 else
    y = gamma (x) ;
 end
```

&& (logical AND) || (logical OR)   ˜ (logical (NO

```
switch units
  case 'length'
    disp ('meters')
  case 'volume'
    disp ('liters')
  case 'time'
    disp ('seconds')
  otherwise
    di sp ('I give up')
end
```

```
x=1:100;  s=0;
for j=find (isprime (x) )
   s = s+ x(j);
end

Finds sum of all primes less than 100
( alternative sum ( f ind ( iprime (1 : 10 0 ) ) ) ;
                                                        )

n=0
while x>1
   x=x/2;
   n=n+1
    i f ( n> 5 0, break, end
end
```

Vectorization

x=1:M

f(x)=sin((x-1)/(2*pi))

[C,R] =meshgrid(c,r);

    rows of C are copies of c and columns of R are copies of r

    example r=[0 1 2] c=[0 1]

    h=`R.^2 + C.^2`yields

    01

    12

    45

    preallocate arrays f=zeros(1024);

Interactive I/O

- disp(A); displays matrix A on screen

- t=input('message');

- n=str2num(t)

Functions by Category

The tables below list all functions in the Image Processing Toolbox by category. The tables include a few functions in MATLAB that are especially useful for image processing, such as `imread`, `imfinfo`, and `imwrite`.

Image Display	
colorbar	Display colorbar. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
getimage	Get image data from axes
image	Create and display image object. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
imagesc	Scale data and display as image. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
immovie	Make movie from multiframe indexed image
imshow	Display image
montage	Display multiple image frames as rectangular montage
subimage	Display multiple images in single figure
truesize	Adjust display size of image
warp	Display image as texture-mapped surface
zoom	Zoom in and out of image or 2-D plot. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)

Image File I/O	
dicominfo	Read metadata from a DICOM message
dicomread	Read a DICOM image
imfinfo	Return information about image file. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
imread	Read image file. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
imwrite	Write image file. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)

Spatial Transformations

checkerboard	Create checkerboard image
findbounds	Find output bounds for spatial transformation
fliptform	Flip the input and output roles of a TFORM structure
imcrop	Crop image
imresize	Resize image
imrotate	Rotate image
interp2	2-D data interpolation. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
imtransform	Apply 2-D spatial transformation to image
makeresampler	Create resampling structure
maketform	Create geometric transformation structure
tformarray	Geometric transformation of a multi-dimensional array
tformfwd	Apply forward geometric transformation
tforminv	Apply inverse geometric transformation

Pixel Values and Statistics

corr2	Compute 2-D correlation coefficient
imcontour	Create contour plot of image data
imfeature	Compute feature measurements for image regions
imhist	Display histogram of image data
impixel	Determine pixel color values
improfile	Compute pixel-value cross-sections along line segments
mean2	Compute mean of matrix elements
pixval	Display information about image pixels
regionprops	Measure properties of image regions
std2	Compute standard deviation of matrix elements

Image Analysis

edge	Find edges in intensity image
qtdecomp	Perform quadtree decomposition
qtgetblk	Get block values in quadtree decomposition
qtsetblk	Set block values in quadtree decomposition

Image Arithmetic	
imabsdiff	Compute absolute difference of two images
imadd	Add two images, or add constant to image
imcomplement	Complement image
imdivide	Divide two images, or divide image by constant.
imlincomb	Compute linear combination of images
immultiply	Multiply two images, or multiply image by constant
imsubtract	Subtract two images, or subtract constant from image

Image Enhancement	
histeq	Enhance contrast using histogram equalization
imadjust	Adjust image intensity values or colormap
imnoise	Add noise to an image
medfilt2	Perform 2-D median filtering
ordfilt2	Perform 2-D order-statistic filtering
stretchlim	Find limits to contrast stretch an image
wiener2	Perform 2-D adaptive noise-removal filtering

Image Registration	
cpcorr	Tune control point locations using cross-correlation
cp2tform	Infer geometric transformation from control point pairs
cpselect	Control point selection tool
cpstruct2pairs	Convert CPSTRUCT to valid pairs of control points
normxcorr2	Normalized two-dimensional cross-correlation

Linear Filtering	
conv2	Perform 2-D convolution. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
convmtx2	Compute 2-D convolution matrix
convn	Perform N-D convolution. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
filter2	Perform 2-D filtering. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
fspecial	Create predefined filters
imfilter	Multidimensional image filtering

Linear 2-D Filter Design

freqspace	Determine 2-D frequency response spacing. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
freqz2	Compute 2-D frequency response
fsamp2	Design 2-D FIR filter using frequency sampling
ftrans2	Design 2-D FIR filter using frequency transformation
fwind1	Design 2-D FIR filter using 1-D window method
fwind2	Design 2-D FIR filter using 2-D window method

Image Transforms

dct2	Compute 2-D discrete cosine transform
dctmtx	Compute discrete cosine transform matrix
fft2	Compute 2-D fast Fourier transform. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
fftn	Compute N-D fast Fourier transform. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
fftshift	Reverse quadrants of output of FFT. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
idct2	Compute 2-D inverse discrete cosine transform
ifft2	Compute 2-D inverse fast Fourier transform. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
ifftn	Compute N-D inverse fast Fourier transform. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
iradon	Compute inverse Radon transform
phantom	Generate a head phantom image
radon	Compute Radon transform

Neighborhood and Block Processing

bestblk	Choose block size for block processing
blkproc	Implement distinct block processing for image
col2im	Rearrange matrix columns into blocks
colfilt	Perform neighborhood operations using columnwise functions
im2col	Rearrange image blocks into columns
nlfilter	Perform general sliding-neighborhood operations

Morphological Operations (Intensity and Binary Images)	
conndef	Default connectivity array
imbothat	Perform bottom-hat filtering
imclearborder	Suppress light structures connected to image border
imclose	Close image
imdilate	Dilate image
imerode	Erode image
imextendedmax	Extended-maxima transform
imextendedmin	Extended-minima transform
imfill	Fill image regions
imhmax	H-maxima transform
imhmin	H-minima transform
imimposemin	Impose minima
imopen	Open image
imreconstruct	Perform morphological reconstruction
imregionalmax	Regional maxima of image
imregionalmin	Regional minima of image
imtophat	Perform tophat filtering
watershed	Find image watershed regions

Morphological Operations (Binary Images)	
applylut	Perform neighborhood operations using lookup tables
bwarea	Area of objects in binary image
bwareaopen	Binary area open; remove small objects
bwdist	Distance transform
bweuler	Euler number of binary image
bwfill	Fill background regions in binary image
bwhitmiss	Binary hit-miss operation
bwlabel	Label connected components in 2-D binary image
bwlabeln	Label connected components in N-D binary image.
bwmorph	Perform morphological operations on binary image
bwpack	Pack binary image
bwperim	Find perimeter of objects in binary image
bwselect	Select objects in binary image

bwulterode	Ultimate erosion
bwunpack	Unpack a packed binary image
imregionalmin	Regional minima of image
imtophat	Perform tophat filtering
makelut	Construct lookup table for use with `applylut`

Structuring Element (STREL) Creation and Manipulation	
getheight	Get the height of a structuring element
getneighbors	Get structuring element neighbor locations and heights
getnhood	Get structuring element neighborhood
getsequence	Extract sequence of decomposed structuring elements
isflat	Return true for flat structuring element
reflect	Reflect structuring element
strel	Create morphological structuring element
translate	Translate structuring element

Deblurring	
deconvblind	Restore image using blind deconvolution
deconvlucy	Restore image using accelerated Richardson-Lucy algorithm
deconvreg	Restore image using Regularized filter
deconvwnr	Restore image using Wiener filter
edgetaper	Taper the discontinuities along the image edges
otf2psf	Convert optical transfer function to point-spread function
psf2otf	Convert point-spread function to optical transfer function

Array Operations	
circshift	Shift array circularly
padarray	Pad an array

Region-Based Processing	
roicolor	Select region of interest, based on color
roifill	Smoothly interpolate within arbitrary region
roifilt2	Filter a region of interest
roipoly	Select polygonal region of interest

Colormap Manipulation

`brighten`	Brighten or darken colormap. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
`cmpermute`	Rearrange colors in colormap
`cmunique`	Find unique colormap colors and corresponding image
`colormap`	Set or get color lookup table. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
`imapprox`	Approximate indexed image by one with fewer colors
`rgbplot`	Plot RGB colormap components. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)

Color Space Conversions

`hsv2rgb`	Convert HSV values to RGB color space. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
`ntsc2rgb`	Convert NTSC values to RGB color space
`rgb2hsv`	Convert RGB values to HSV color space. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
`rgb2ntsc`	Convert RGB values to NTSC color space
`rgb2ycbcr`	Convert RGB values to YCbCr color space
`ycbcr2rgb`	Convert YCbCr values to RGB color space

Image Types and Type Conversions

`dither`	Convert image using dithering
`double`	Convert data to double precision. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
`gray2ind`	Convert intensity image to indexed image
`grayslice`	Create indexed image from intensity image by thresholding
`graythresh`	Compute global image threshold using Otsu's method
`im2bw`	Convert image to binary image by thresholding
`im2double`	Convert image array to double precision
`im2mis`	Convert image to Java MemoryImageSource
`im2uint16`	Convert image array to 16-bit unsigned integers
`im2uint8`	Convert image array to 8-bit unsigned integers
`ind2gray`	Convert indexed image to intensity image
`ind2rgb`	Convert indexed image to RGB image
`isbw`	Return true for binary image

`isgray`	Return true for intensity image
`isind`	Return true for indexed image
`isrgb`	Return true for RGB image
`label2rgb`	Convert a label matrix to an RGB image
`mat2gray`	Convert matrix to intensity image
`rgb2gray`	Convert RGB image or colormap to grayscale
`rgb2ind`	Convert RGB image to indexed image
`uint16`	Convert data to unsigned 16-bit integers. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)
`uint8`	Convert data to unsigned 8-bit integers. (This is a MATLAB function. See the online MATLAB Function Reference for its reference page.)

Toolbox Preferences	
`iptgetpref`	Get value of Image Processing Toolbox preference
`iptsetpref`	Set value of Image Processing Toolbox preference