Practical Session 1

ICBV171

The Interdisciplinary Computational Vision Laboratory
Computer Science Department
Ben-Gurion University of the Negev
Administrative Information

Course Website
http://www.cs.bgu.ac.il/~icbv181

Pay attention to the announcements!

Read the reading material (yes, really!)

Lab Website
http://icvl.cs.bgu.ac.il

Office Hours: coordinated via email boazar@bgu.ac.il
MATLAB

- Proprietary scripting language.
- Everything is a matrix.
- Highly optimized.
- Tightly integrated.

Easy to learn and use
Let’s see your homework
Learning MATLAB

• Google is your friend
  MATLAB has a vibrant and active developer community.

• Additional resources:
  MATLAB documentation (pay attention to version numbers)
  MATLAB Central – user forum
  Stack Overflow – go easy on the copy/paste
Step 1: Write code

Step 2: Realize that 50% of the code can be replaced with a single line of code.

Step 3: Update code, Go to step 2.
MATLAB development cycle

Convert a color image to grayscale:

% load an image
I=imread('trailer.jpg');
% prepare a matrix to hold grayscale version of image
Ig=zeros(size(I,1),size(I,2));
% collect weighted of sum R,G,B pixels into grayscale image
for x=1:size(I,1)
    for y=1:size(I,2)
        Ig(x,y)=0.2989*I(x,y,1)+...
              0.5870*I(x,y,2)+...
              0.1140 * I(x,y,3);
    end
end
% convert doubles back to uint8 matrix
Ig = uint8(Ig);
% display grayscale image
imshow(Ig);
MATLAB development cycle

Convert a color image to grayscale:

% load an image
I = imread('trailer.jpg');
% prepare a matrix to hold grayscale version of image
Ig = zeros(size(I,1), size(I,2));
% collect weighted of sum R,G,B pixels into grayscale image
for x = 1:size(I,1)
    for y = 1:size(I,2)
        Ig(x,y) = 0.2989*I(x,y,1) + ...
                    0.5870*I(x,y,2) + ...
                    0.1140 * I(x,y,3);
    end
end
% convert doubles back to uint8 matrix
Ig = uint8(Ig);
% display grayscale image
imshow(Ig);
MATLAB development cycle

Convert a color image to grayscale:

% load an image
I = imread('trailer.jpg');
% collect weighted of sum R,G,B pixels into grayscale image
Ig = 0.2989*I(:,:,1) + 0.5870*I(:,:,2) + 0.1140*I(:,:,3);
% convert doubles back to uint8 matrix
Ig = uint8(Ig);
% display grayscale image
imshow(Ig);
MATLAB development cycle

Convert a color image to grayscale:

% load an image
I=imread('trailer.jpg');
% collect weighted of sum R,G,B pixels into grayscale image
Ig = 0.2989*I(:,:,1)+0.5870*I(:,:,2)+0.1140*I(:,:,3);
% convert doubles back to uint8 matrix
Ig = uint8(Ig);
% display grayscale image
imshow(Ig);

Oh, wait!
There’s a function for that
MATLAB development cycle

Convert a color image to grayscale:

% load an image
I=imread('trailer.jpg');
% convert to grayscale
Ig=rgb2gray(I);
% display grayscale image
imshow(Ig);
MATLAB development cycle

Convert a color image to grayscale:

% load an image
I=imread('trailer.jpg');
% convert to grayscale
Ig=rgb2gray(I);
% display grayscale image
imshow(Ig);
MATLAB development cycle

Convert a color image to grayscale:

```matlab
% load an image and display as grayscale
imshow(rgb2gray(imread('trailer.jpg')));
```

Conclusion:
If your code is more than 1 line long – you’re probably doing something wrong!
MATLAB Basics

Defining variables:

\[
\begin{align*}
A &= [1 \ 2 \ 3 \ 4] \\
B &= [1 \ 2 ; \ 3 \ 4] \\
C(5,5) &= 1 \\
D &= 1:4 \\
E &= 1:2:10 \\
R &= \text{rand}(10)
\end{align*}
\]

Try it!

Variable indexing

\[
\begin{align*}
A(4) \\
B(:,2) \\
B(3) \\
C(1:5,5) \\
R(5,5)
\end{align*}
\]

Try it!
MATLAB Basics

Generating matrices:

I=eye(5)
O=ones(5)
Z=zeros(5)
R=rand(5)
Ri=randi(10,5,5)

Try it!

Primitive functions

max(Ri)
min(Ri(:))
sort(Ri(:))
sqrt(4)
sqrt(1:4)
sqrt(Ri)

Try it!
**Math pitfalls**

```matlab
clear all
A=ones(4)*2; % note the ‘;’
B=ones(4)*4;
A+B
B-A
A*B % did this work as expected?
A^2 % what about this?
```

**Try it!**

**Primitive functions**

```matlab
max(Ri)
min(Ri(:))
sort(Ri(:))
sqrt(4)
sqrt(1:4)
sqrt(Ri)
```

**Try it!**
MATLAB Basics

Element-wise operations

\[ A = \text{ones}(4) \times 2; \]
\[ B = \text{ones}(4) \times 4; \]
\[ A.*B \]
\[ B./A \]
\[ B.^2 \]

Try it!

Matrix wizardry

\[ A = \text{magic}(3) \]
\[ B = [15;15;15]; \]
\[ x = A \backslash B \]
\[ B \times x \]
\[ x = B / A \]
\[ A \times x \]

Try it!