INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA
MESTRADO EM ENGENHARIA INFORMÁTICA E DE COMPUTADORES
MESTRADO EM ENGENHARIA INFORMÁTICA E MULTIMÉDIA
IMAGE PROCESSING AND BIOMETRICS

First Laboratory Class - MATLAB/OCTAVE
1st semester, 2021/2022 (October, 22)

---

1. **Work teams and software.**
   Setup the work teams (two to three students).

   Install software 1 (MATLAB) and software 2 (GNU Octave 6.3 or earlier).

2. **MATLAB environment - setup, documentation, examples, demos, and apps.**

   (i) Install and configure the MATLAB environment (with the Signal Processing Toolbox, the Image Processing Toolbox, and the Code Generation Tools) and check for their proper functioning.

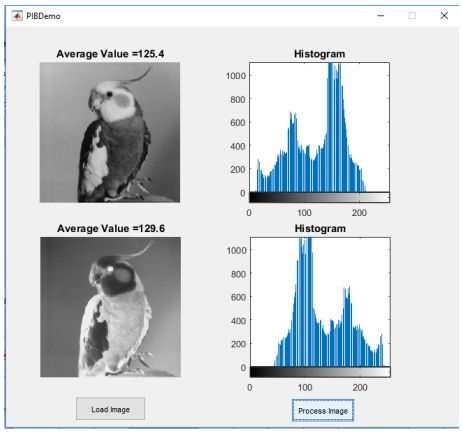
   (ii) Explore the MATLAB Image Processing Toolbox features, at `https://www.mathworks.com/products/image.html` and watch the video available on this webpage.

   (iii) Check for the proper functioning of the four *.m files on the MATLAB demos named Demo1.zip and Demo2.zip, available on the Moodle page.

   (iv) Check for the proper functioning of the Graphical User Interface (GUI) named PIBDemo, as displayed in Figure 1(a). Take a look at the code on the `PIBDemo.m` and `PIBDemo.fig` modules by typesetting the following commands on the MATLAB command window: `edit PIBDemo.m` and `guide PIBDemo.fig`. The MATLAB `guide` tool, depicted in Figure 1(b), allows you to develop graphical user interfaces with common graphical controls.

   (v) Check on the functioning of the executable version of the PIBDemo GUI, developed by the `Application Compiler Tool`, under the `APPS` tab, as depicted in Figure 2(a).

   (vi) Take a look at the different demos provided by MATLAB, available for different apps, as depicted in Figure 2(b). For instance, explore the `Image Segmenter` and the `Bit Error Rate Analysis` demos.



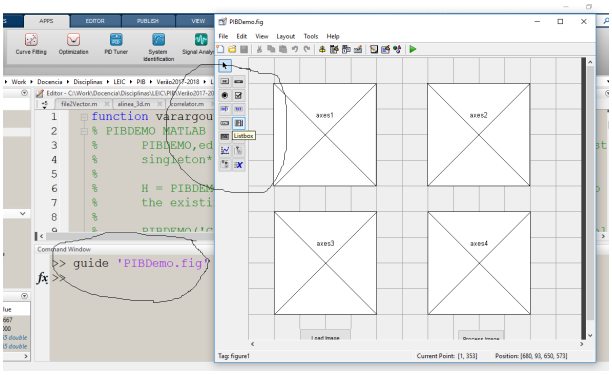
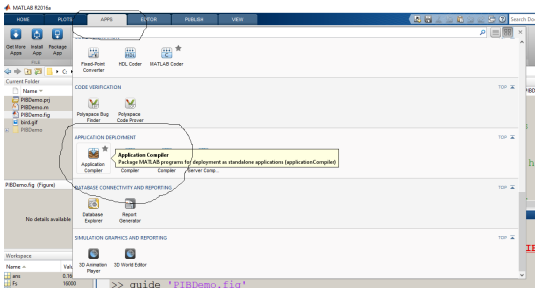


Figure 1: (a) The PIBDemo Graphical User Interface (GUI) (b) The guide tool to develop GUI applications.



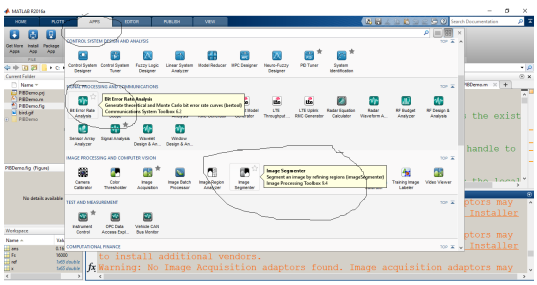


Figure 2: (a) The Application Compiler Tool (b) Demos for some APPS available with MATLAB.

3. **Resources and Tools.**
   Check on the features of the following resources and tools for Digital Image Processing (DIP).

   Some image databases:

   - **Image Processing Place web page**, `http://www.imageprocessingplace.com/`, `http://www.imageprocessingplace.com/root_files_V3/image_databases.htm`;
   - **The Waterloo fractal coding and analysis group**, and the **Waterloo Bragzone**, `http://links.uwaterloo.ca/Repository.html`, `http://links.uwaterloo.ca/oldwebsite/bragzone.base.html`

   Digital image processing tools in Java:

   - **ImageJ**, Image Processing and Analysis in Java, `http://imagej.nih.gov/ij/index.html`
   - **JVT - Java Vision Toolkit**, `http://javavision.sourceforge.net/`
   - **Java DIP - Digital Image Processing**, `http://www.tutorialspoint.com/java_dip/index.htm`

   Digital image processing tools in C#:

   - **AForge.NET**, Open Source C# Framework, `http://www.aforgenet.com/framework/`
   - **Accord Framework .NET**, `http://accord-framework.net/`
   - **Image Processing Lab in C#**, `http://www.codeproject.com/Articles/9727/Image-Processing-Lab-in-C`

4. **[MATLAB - Exercise 1]**.

   (a) Explain, in detail, the DIP actions taken by the four *.m files on the MATLAB demos named Demo1.zip and Demo2.zip, available on the Moodle page.

   (b) Write the MATLAB function `dip_analysis` that performs the following actions on a given image:
   (i) counts the number of distinct pixels and the total of pixels;
   (ii) finds the most frequent pixel value and the least frequent pixel value;
   (iii) computes the brightness, contrast, and entropy values.

   (c) Report some experimental results of the MATLAB function `dip_analysis`, for the 256×256 Lena, Bird, and Boat images, available at `http://links.uwaterloo.ca/Repository.html`