IB Paper 8: Photo Editing

Lecture 1: Program Framework and Cropping

N G Kingsbury (given by J Lasenby in 2016)

Signal Processing Group,
Engineering Department,
Cambridge, UK

Easter 2016

Contents

- *Lecture 1:* Sections 1 & 2: Program framework and Cropping.
- *Lecture 2:* Sections 3 & 4: Resizing and Rotating.
- *Lecture 3:* Sections 5 & 6: Morphing and Colour.
- *Lecture 4:* Sections 7 & 8: Histograms, Lighting and Filtering.
- *Lecture 5:* Section 9: Enhancing.
- *Lecture 6:* Examples Class

Second and Third Sections of the Course

B – Image Features and Matching Course

Roberto Cipolla

C – Image Searching and Modelling using Machine Learning Methods

Roberto Cipolla

Recommended Textbooks

- A K Jain, *Fundamentals of Digital Image Processing*, Prentice-Hall, 1989.
- R C Gonzalez and R E Woods, *Digital Image Processing*, Addison Wesley, 1992.
- M Petrou and P Bosdogianni, *Image Processing: The Fundamentals*, John Wiley, 1999.

Matlab code for the Photo Editor is downloadable from NGK's website:
www-sigproc.eng.cam.ac.uk/∼ngk in the section *Downloadable teaching material*.

Introduction

- Aim: to develop a photo-editor in Matlab
- First task: produce an image display framework and graphical user interface (GUI) to allow the functions to be called and controlled, and to show results.
- Choose Matlab because: allows quite complicated image processing operations to be implemented with relatively small amounts of code, and it has the basic functions for providing a suitable GUI for the user to interact with the program.

The Main Script File: ph_edit

- The code for the main script file ph_edit.m is shown in fig. 1.2. in the notes.
- No variables are passed as arguments when script files are called.
- For GUIs with controls and buttons, it is simplest if calls are created to script files.
- Now look at the code to see how the GUI is set up.

Opening an input file: `ph_openfile`

- `ph_openfile` is a script called by the menu item **Open 'Before'** (and also at the end of `ph_edit` – although there are a few problems with this).
- `uigetfile` is used to open a GUI window to allow selection of an input image file (.tif, or .jpg, but could be others). Directory path and filename then concatenated into `infile`.
- `imread` reads in image into `xui`. If no output image `yui`, `xui` is read into `yui`.
- `xui` and `yui` are displayed as **Before** and **After** via `showimages`.

Saving an output file: ph_savefile

- Script for saving files is given in notes: it is called by the menu item **Save 'After'** and also when **Close Editor** is activated.
- Code creates an outfile by adding an _a to the input filename (if the ending is not already _a)
- uiputfile then called to allow user to confirm, change or cancel.
- imwrite then saves the **After** image yui as .tif or .jpg.

Displaying the images: `showimage`

- Fig 1.5 in the notes shows the code for displaying images and their colour histograms.
- `newbefore` is a variable which ensures (if it is 0) that the **Before** image is not updated – this speeds things up.
- `subplot` and `image` Matlab functions are used to plot the RGB images.
- Matlab functions `reshape`, `barcolour`, `hist` are used to plot the histograms.
- Zoom is enabled.

Other simple operations within: `ph_edit`

Look at the top 5 menu options:

- **Open 'Before'**: via a call to `ph_openfile`
- **Reopen 'Before'**: re-reads from input file via: `xui = imread(infile); showimages`
- **Save 'After'**: via a call to `ph_savefile`
- **Copy 'Before' to 'After'**: via `yui=xui; newbefore=0; showimages`
- **Copy 'After' to 'Before'**: via `xui=yui; newbefore=1; showimages`

Cropping the image: ph_crop

ph_crop is the first of the 9 script files that perform the main operations within the Photo Editor. Code is given in Fig 2.1.

- Scripts are all controlled by mode: a string variable in the main workspace.
- All options are in the same script file and are selected by mode using the switch syntax in Matlab.
- **Note**: switch executes only the first matching case, there is no need for **break** statements.
- The crop area can be selected via entering coordinates of a rectangle or via the cursor (using Matlab **zoom**).

Summary

- Section 1 of the notes outlines how the Photo Editor works – ie making use of the GUI facilities in Matlab and adopting a script approach with the **switch** and **mode** functionalities.
- Section 2: The first of the 9 main script files, `ph_crop`, was examined, both for its use of switch/mode and for its intrinsic working.

J. Lasenby (Easter 2016)