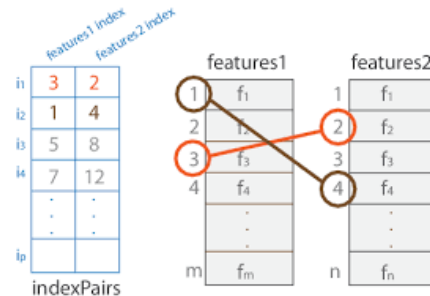
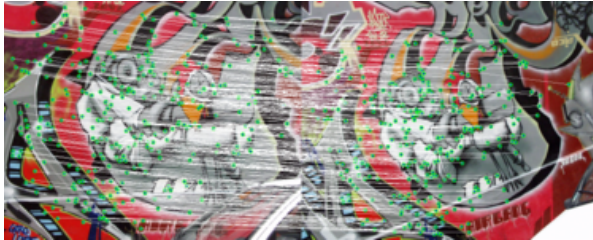


## Machine Learning for Computer Vision

### Coursework on image matching [50% mark]



**Release on 3 Mar 2017, the report due on 2 May 2017 (midnight)**

The course work requires Matlab programming. In all questions, you can use any existing toolbox/code, unless specified.

#### Submission instructions:

One joint report by each pair

Page limit: **4** A4 pages per report with 10 font size (use the IEEE standard double column paper format, either in MS word or latex).

[http://www.pamitc.org/cvpr16/files/egpaper\\_for\\_review.pdf](http://www.pamitc.org/cvpr16/files/egpaper_for_review.pdf)

<http://www.pamitc.org/cvpr16/files/cvpr2016AuthorKit.zip>

Give insights, discussions, and reasons behind your answers, on the scope of lectures. **Quality and completeness of discussions within the page limit** will be marked. Include formulas where appropriate, results presented in figures and their discussion.

Source code, such that the experiments and the results can be reproduced, is not mandatory, however it can help disambiguate in case the results are arguable: this can go to appendices, which do not count for the page limit.

Submit the report **in pdf** through the Blackboard system. No hard copy is needed. Write your full names and CID numbers on the first page.

If you have questions, please contact

Mr. Guillermo Garcia-Hernando ([g.garcia-hernando@imperial.ac.uk](mailto:g.garcia-hernando@imperial.ac.uk))

Dr. Vassileios Balntas ([v.balntas@imperial.ac.uk](mailto:v.balntas@imperial.ac.uk))

Collect a sequence of pictures (we call it **FD**) of the same scene (e.g. robots exhibition in science museum). At least 3 pictures should be taken by changing the camera position by 20cm in horizontal direction (see stereo examples in the lecture notes). Another sequence (we call it **HG**) of 3 pictures should be taken by changing the zoom (e.g. factor 1.5) and slightly rotating the camera (e.g. 10-20 degree) but keeping exactly the same location of the camera. Try to find out the focal length of your camera for each image, typically between 18-55mm (varies with zoom), otherwise assume a length.

Matlab implementation code should be included in Appendix.

You can use Boat and Tsukuba sequence as a data with ground truth for development before testing on your images.

[http://www.robots.ox.ac.uk/~vgg/research/affine/det\\_eval\\_files/boat.tar.gz](http://www.robots.ox.ac.uk/~vgg/research/affine/det_eval_files/boat.tar.gz)

<http://vision.middlebury.edu/stereo/data/scenes2001/>

## Q1. Matching

### 1) Manual

- a) Implement a manual method e.g. loading image in Matlab (imread, imshow) and by carefully clicking with mouse pointer (ginput) get coordinates of interest points in image A and then click the same points, in the same order, in image B.

### 2) [10] Automatic

- a) Implement interest point detector (e.g. Harris). Approximate derivatives with convolution  $[-1 \ 0 \ 1]$  and blurring  $[0.03 \ 0.105 \ 0.222 \ 0.286 \ 0.222 \ 0.105 \ 0.03]$ . The method should output a  $2 \times N$  matrix which is a list of coordinates (x,y) of points.
- b) Implement a method that given the  $2 \times N$  matrix with coordinates from Q1.2.a returns descriptor vectors in a matrix of size  $N \times M$  where the rows are descriptors of dimensionality M (the matrix may have less rows than N). A descriptor can be calculated from a  $32 \times 32$  patch cropped at a given point. A simple colour histogram can be implemented, or a sampled patch to a lower dimensionality (e.g.  $M=121$ ).
- c) Implement a method that performs nearest neighbour matching of descriptors. Given two matrices containing descriptors from image A and B, the method should output a list of correspondences between images A and B, such that a descriptor from image A is a nearest neighbour to a descriptor from B and vice versa.

### 3) [10] Transformation estimation

- a) Implement a method for estimating a homography matrix given a set of corresponding point coordinates. The method should take as an input two 2D vectors resulting from Q1.2.c or Q1.1.a, which contain coordinates of corresponding points in images A and B.
- b) Similarly to Q1.3.a, implement a method for estimating a fundamental matrix given a set of corresponding point coordinates.
- c) Implement a method for projecting point coordinates from image B to A given Homography between the two images. Implement a method for calculating average distance in pixels between the original points in A and the ones projected from B. This error can be interpreted as homography accuracy **HA**.

- d) Implement a method for calculating epipolar line given point coordinates and a Fundamental matrix between two images. Display that line on an image in Matlab.

## Q2. Image Geometry

### 1) [15] Homography (use images HG)

- a) Find interest points in one image by using method from Q1.1 or Q1.2.a. Reduce the size of the image by a factor of 2 and run detector again. Compare the interest points obtained in these two cases using **HA** error.
- b) Using method from Q1.3.a estimate homography from the manually established correspondences with Q1.1 and compare to the homography from the list of correspondences obtained automatically with Q1.2. Use function from Q1.3.c to visualise and validate your homographies. Analyse and compare geometric transformation parameters that can be derived from the two homographies.
- c) Estimate homography from different number of correspondences from Q1.2 starting from the minimum number up to the maximum number of available pairs. Report and discuss **HA** for different number of correspondences. Find the number of outliers in your list of automatic correspondences and explain your approach to that.

### 2) [15] Stereo Vision (use images FD)

- a) Estimate fundamental matrix using list of correspondences from Q1.1 or Q1.2.a.
- b) Calculate the epipoles for images A and B. Show epipolar lines and epipoles on the images if possible.
- c) Calculate disparity map between images A and B.
- d) Calculate and display depth map (e.g. in a 3D plot).
- e) Change the focal length by 2mm, repeat Q2.2.d and compare. Add small random noise (e.g. Gaussian with max 2 pixel) to the disparity map, repeat Q2.2.d and compare.
- f) Extra challenge: perform stereo rectification of a pair of your images.

---

Follow general principles for writing technical report on data processing.

Select relevant results for reporting rather than everything that Matlab gives, also use clear presentation style consistently across the report. The important results should be in the report, not just in the appendix.

The space used efficiently, figures and labels of appropriate size, readable, informative with the appropriately chosen presentation style. Matlab screenshots, especially if too low resolution, don't show well unless font size, background and colours adjusted.

Figures and tables have captions and are referred to in the text. When discussing results the main points should be made clear, identifying and concisely explaining interesting observations and discussing possible causes. These should come from the results rather than reporting what is known in general without referring to the results.

No need to put standard formulas, explain the algorithms in detail or copy figures from other sources. References to particular lecture slides or publications/webpages are enough in such cases, however short explanations of specific terms or parameters used in figures or text are needed.