Machine Learning for Computer Vision: Coursework 2 - Image Matching

Tormento, Marion
mjt17, CID:01401447
marion.tormento17@imperial.ac.uk

McLaughlin, Edward
em215, CID:01092693
edward.mclaughlin17@imperial.ac.uk

1 Matching

1.2 Finding points automatically

In order to find interest points automatically, several corner detectors were implemented. The performance of each algorithm was tested against toolbox corner detectors available on OpenCV. As SURF and SIFT corner detectors are not freely available on OpenCV, they were not implemented. Once relevant interest points had been located, several descriptors were used to characterise the corner points. The performance of each descriptor was assessed on its ability to correctly match corresponding interest points using a nearest neighbour algorithm.

1.2.1 Interest points and descriptors:

Initially, Harris [1] and Shi-Tomasi [3] interest point detectors were implemented. These detectors differ only in their evaluation of R - the determination of whether a given point is considered a corner. In both cases, the threshold of the R value was set to the 99-th percentile value in order to obtain enough interest points in a reasonable computational time. Subsequently, a Features from Accelerated Segment Test (FAST) [4] feature detector was implemented.

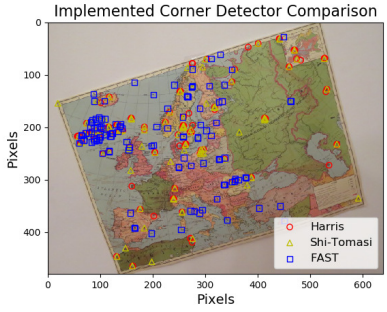
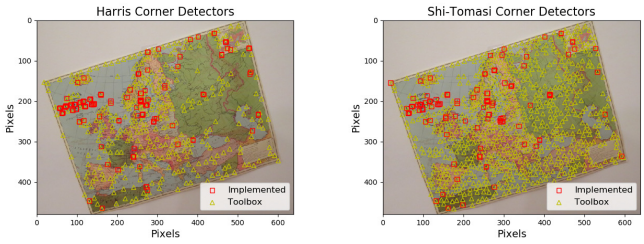


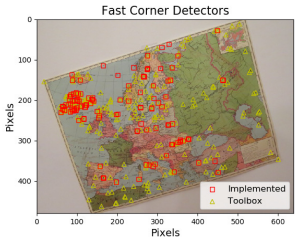
Figure 1. Comparison of corner points detected by implemented solutions. Harris - 138, Shi-Tomasi - 121, FAST - 125

The corner points detected from these three detectors are shown in Figure 1. The Harris and Shi-Tomasi detectors find very similar points whereas the FAST detector, as might be expected, picks up the larger black dots on the map representing large cities. The comparison of the corner points found by the detectors implemented by the authors and the toolboxes available on OpenCV is shown in Figure 2. Further results on corner point detection are documented in the Appendix. It is evident that the toolbox detectors find considerably more corner points, most likely due to the algorithms implemented taking only the 99th percentile for maxima suppression. However, finding many more points appeared detrimental as it increased the likelihood of interest points being mismatched. Three types of descriptor were implemented: colour detector (RGB), Histogram of Oriented Gradients (HOG) and RGBHOG. RGBHOG is constructed by concatenating HOGs for each colour channel in the window around the interest point. The parameters used for each interest point detector and descriptor are detailed in the Appendix.



(a) Harris: Implemented - 138 Toolbox - 380

(b) Shi-Tomasi: Implemented - 121 Toolbox - 768



(c) FAST: Implemented - 125 Toolbox - 297

Figure 2. Comparison of corner points detected by implemented solutions and openCV algorithm.

1.2.2 Matching interest points across images

To match interest points, a nearest neighbour search algorithm was implemented on the histograms of the descriptors using the Sum of Squared Differences (SSD) as the metric. To maximize the chances of returning true positive matches, the following three strategies were implemented: 1) the ratio technique, first presented by D. Lowe [2], was implemented whereby the quality of a match is determined by the ratio of the match and the second best match. Unfortunately, this technique did not return interesting results; 2) to avoid clustering of interest point which negatively affects the matching performance, only the best match of each cluster is saved as an interest point; 3) the final matches are ordered in a list such that the best matches are easily accessed.

When testing the descriptors, it is was found that RGB is best suited to pictures which are rotated while HOG is best suited to pictures which are zoomed in or shifted. Figure 6 shows a subset of the matches interest points. In this case, the RGB descriptor was used - it is evident that the outliers (black) are false matches due to

the similar hue in the two locations on the map.

1.3 Transformation estimation

1.3.1 Homography Matrix (HM) and Accuracy (HA)

The HM is produced from a minimum of four corresponding points. It is calculated by solving the simultaneous equations given in eqn 1 where x and $x\prime$ are the corresponding points in the first and second image respectively and H denotes the homography matrix. These are solved using Simultaneous-Value Decomposition (SVD). The accuracy is calculated as the average distance between the actual point and the projected point in the second image. The projection of interest points in the first image to the second image is visualised in Figure 4.

1.3.2 Fundamental Matrix (FM) and Accuracy (FA)

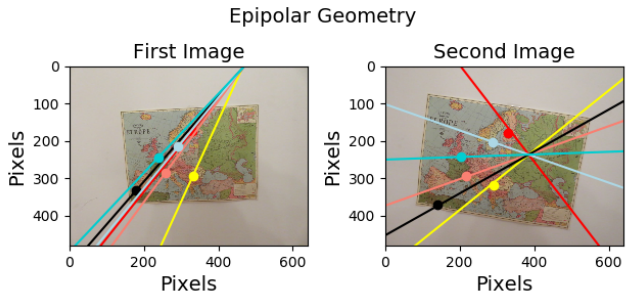


Figure 3.

The FM is determined by using the 8 point algorithm which uses SVD to solve the set of linear equations given by eqn. 2 where F denotes the FM. A minimum of eight points are needed to adequately determine the FM. The accuracy is calculated as the average orthogonal distance between the calculated epipolar lines and the actual points in the corresponding image (see Figure 3).

$$x\prime H = x \tag{1}$$

$$x\prime F x = 0 \tag{2}$$

2 Image Geometry

2.1 Homography

2.1.1 Homography accuracy for reduced image sizes

A set of images of a rotated map was used for this question. The detector used was the implemented Harris Corner detector and the results were compared to the openCV version. All types of descriptor are normalised and tested using dynamic window sizing for consistent results. The homography matrix is computed between the original image and one of a reduced size. The Homography Accuracy (HA) is given both as average distance in pixels and as a percentage of the minimum of the dimension of the reduced image. The full results can be seen in Table 1. It was found that the HM does not give good accuracy for reduced sized images. An average percentage error of 23%, and 20% for the implemented and OpenCV descriptors respectively was found for a reduction factor of 2. The average percentage error for a reduction factor of 3 was 31% and 22% respectively. The OpenCV algorithm returns more accurate results, but in both cases increasing the reduction factor, decreases the accuracy of the matching. Indeed, the smaller the image, the harder it is to match relevant interest points as the descriptor information is compressed in fewer pixels.

2.1.2 Homography matrix and its validation

The homography matrix is computed for manually and automatically picked interest points. When enough interest points were manually selected (25 or more), the value returned was close to the automatic one, with an error of 5% between the two HMs. The HM was validated graphically in Figure 4 where the majority of the points are well mapped to the second image. The translation of the images into the plane of their respective counterpart renders similar images, further validating the HM. From the two sets of matching points, the geometric transformation parameters could be estimated (eqns. 3 - 5). The real transformation was estimated from image measures on Photoshop. These values confirm an accurate HM.

Table 1. Homography Accuracy in Pixel (Px) and Percentage between Original Image and its reduction by a factor of 2 (RF2) or 3 (RF3)

		RF2		RF3	
Detector	**Descriptor**	**Px**	%	**Px**	%
Implemented	RGB	56	23	42	26
	HOG	59	25	58	36
	RGBHOG	36	15	43	27
OpenCV	RGB	27	11	24	15
	HOG	55	23	38	24
	RGBHOG	72	30	33	21

$$\vec{T_{manual}} = \begin{pmatrix} 29 & -50 & 0 \end{pmatrix}^T, R_{manual} = 12.7^o \tag{3}$$

$$\vec{T_{auto}} = \begin{pmatrix} 36 & -42 & 0 \end{pmatrix}^T, R_{auto} = 10.1^o \tag{4}$$

$$\vec{T_{real}} = \begin{pmatrix} 19 & 20 & 0 \end{pmatrix}^T, R_{real} = 11^o \tag{5}$$

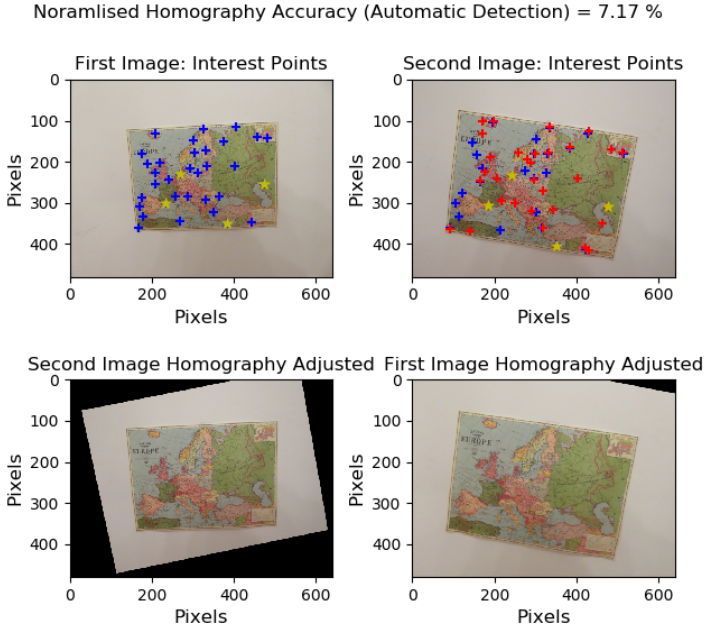


Figure 4. Interest points: used (yellow ⋆) and not used (blue +) for HM calculation. The points from image A projected by the HM to image B are shown red +. The bottom two images are the images transformed to the plane of their respective counterpart.

2.1.3 Dealing with outliers

The implemented Harris corner detector and RGB descriptor are used to answer this question as they returned the best results. They will be compared to the results obtained with openCV embedded method. As it can be seen in Figure 5, no correlation can be made between the number of matching interest points and the HA, although an initial guess would have been that increasing the number of matches improves the HA. This is due to the fact that the matching algorithm returns both correct matches and outliers. The quality of the HM will depend on the proportion of inliers

to outliers - on average the implemented method returns 67% of inliers, against 70% for the toolbox. Figure 6 shows the outliers from matching. These were dealt with by using RANSAC to find the best HM and FM from 4 and 8 points respectively. As a result, the HA is improved to 5.5% for the implemented version, and 6.2% for the toolbox.

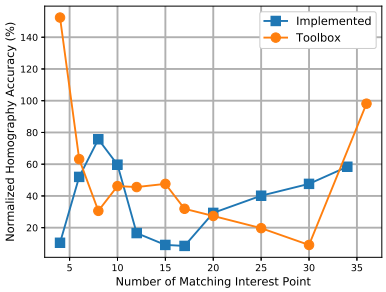


Figure 5. Normalised HA depending on the number of matching interest point used to compute the HM (without RANSAC).

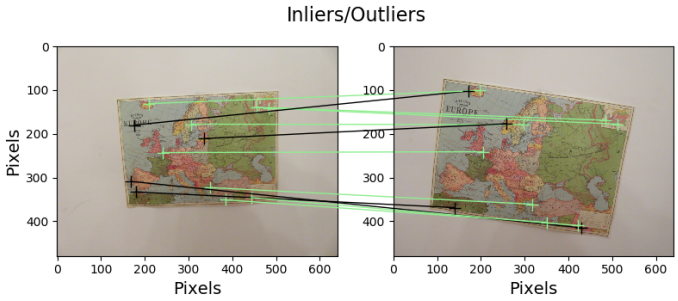


Figure 6. A subset of the matched points having had the HM applied to them. Outliers (black) are defined as matches with greater than 5 pixels distance between the actual and projected points in the second image.

2.2 Stereo Vision

2.2.1 Fundamental Accuracy (FA)

FMs were calculated using manual and automatic interest point detection. The normalised FA were 5.33% and 5.07% respectively. While these results are comparable they rely heavily on careful manual selection of interest points.

2.2.2 Epipoles and epipolar lines

The epipoles (E_L and E_R) and epipolar lines for this set of images are shown in Figure 7. The disparity and depth maps corresponding to the images are shown in Figure 8. In general, the outlines of the shapes are picked up well while the detailed patterns appear noisy. In agreement with the theory, in the disparity map, the objects which are determined to move more appear to be closer in the depth map. The monochromatic background, as expected, creates holes in the disparity and depth maps alike.

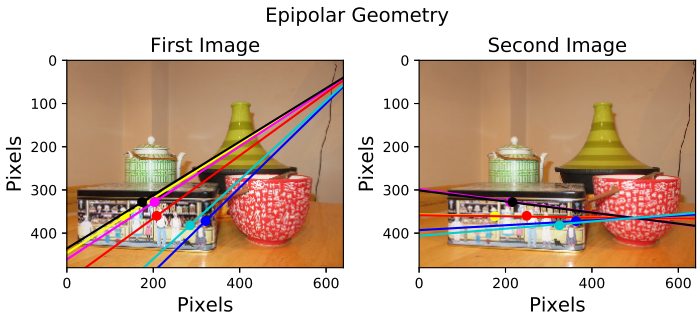


Figure 7. E_L - (698, 4); E_R - (498, 364); Normalised FA - 4.74%

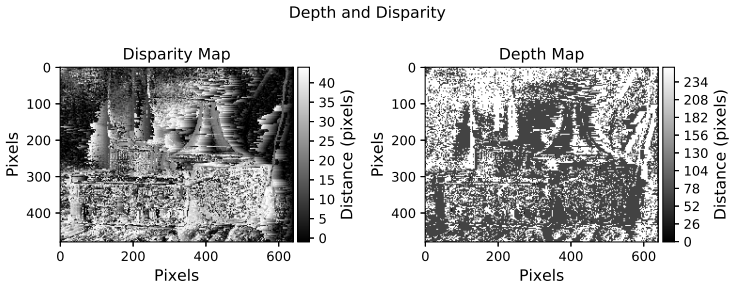


Figure 8. Disparity map (left) and depth map (right) for the image set shown in Figure 7.

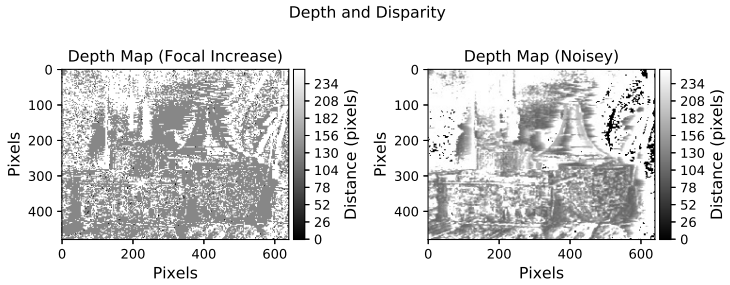


Figure 9. Disparity maps for increased focal length (left) and 2 pixels of added Gaussian noise (right).

2.2.3 Disparity and depth maps

The depth map is linearly and inversely proportional to disparity as shown in eqn 6, where f is the focal length and b is the distance between the camera sensors. As a result, changing the focal length does not change the relative depth of objects in the image, only their absolute distances from the camera lenses. This is evident by comparing Figure 8 (right) and Figure 9 (left). In the latter, the increased focal length increases the perceived distance of the objects from the camera lens.

$$Depth = \frac{fb}{disparity} \tag{6}$$

Introducing Gaussian noise (Figure 9 (right)) smooths out the discontinuities in the depth map. The resultant blurring also causes the objects to appear to be further away on the depth map. This effect may be caused by the uncertainty in the pixel translation introduced by the Gaussian noise.

2.2.4 Stereo rectification

Figure 10 shows the images with their epipolar lines. Figure 10 (right) is the second image projected into the plane of the first image using the HM. While this method is not classical stereo rectification of images, the epipolar lines shown are (almost) horizontal with an epipole far from the image plane, representing that the two images are quasi-coplanar. This method of stereo rectification is not ideal as the second image becomes warped compared with the first.

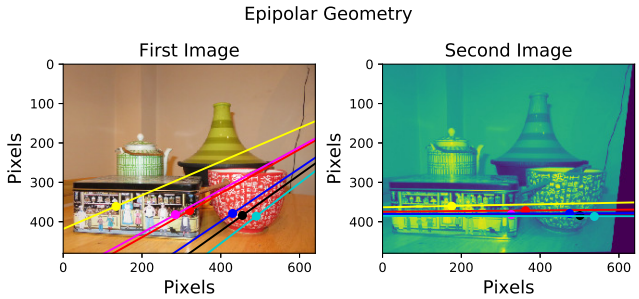


Figure 10. E_L - (1015, -15); E_R - (-1073, 384); Normalised FA - 0.8%

References

[1] C. Harris and M. Stephens. A Combined Corner and Edge Detector. In *Alvey vision conference*, pages 147–151, 1988.

[2] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[3] C. Tomasi and J. Shi. Good Features. *Image (Rochester, N.Y.)*, pages 593–600, 1994.

[4] M. Trajkovii, M. Hedley, M. Trajkovic, and M. Hedley. Fast corner detection. *Image and Vision Computing*, 16(2):75–87, 1998.

3 Appendix

Code

The functions created are available at https://github.com/MarionTormento/MLCV for your information.

Comparison of interest points for each detector

Table 2. Number of interest points detected by each detector.

	Harris		Shi-Tomasi		FAST	
	# Corner Points Detected					
Image (size)	**Implemented**	**ToolBox**	**Implemented**	**ToolBox**	**Implemented**	**ToolBox**
Map 1 (640x840)	129	164	136	407	55	112
Map 2 (640x840)	138	380	121	768	125	297
Computer Room (640x840)	113	94	106	93	187	364
Living Room (640x840)	78	87	94	183	62	131
Tsukuba (384x288)	51	47	52	87	192	353
Art Work (1390x1110)	720	458	672	985	278	969
	# Corner Points Detected per 10,000 pixels					
Image (size)	**Implemented**	**ToolBox**	**Implemented**	**ToolBox**	**Implemented**	**ToolBox**
Map 1 (640x840)	2.4	3.1	2.5	7.6	1.0	2.1
Map 2 (640x840)	2.6	7.1	2.3	14.3	2.3	5.5
Computer Room (640x840)	2.1	1.7	2.0	1.7	3.5	6.8
Living Room (640x840)	1.5	1.6	1.7	3.4	1.2	2.4
Tsukuba (384x288)	4.6	4.2	4.7	7.9	17.4	31.9
Art Work (1390x1110)	4.7	3.0	4.4	6.4	1.8	6.3
Mean	3.0	3.5	2.9	6.9	4.5	9.2
Standard Deviation	1.2	1.8	1.2	4.0	5.8	10.3

Parameters

Harris and Shi-Tomasi corner detector

- σ_I: automatically vary with the size of the image;
- σ_D: set to $1.6\sigma_I$ according to the lecture slides;
- α: set to 0.04 - should be between 0.04 and 0.06 according to online documentation;
- $maxima_{NN}$: number of nearest neighbours when performing non maxima suppression, set to 50 to reduce clustering;
- $maxima_{perc}$: the percentile threshold for the corner point's R value used to qualify corner points for maxima suppression.

FAST detector

- radius: chosen as 3 which optimised the performance and computation time;
- threshold: found empirically for each image as it depends on the distribution of intensities in the image;
- S: varied between 7 and 10 depending on the image.

RGB Color descriptor

- windowSize: Size of the window surrounding the interest point on which the description is performed

Histogram of Oriented Gradient and RGBHOG

- windowSize: Size of the window surrounding the interest point on which the description is performed
- nbBins: number of bins, directly affects the precision of the histogram - ie if there is 36 bins, each bin is of size $\frac{360}{60} = 10^o$.