Practical block I: Landmark detection and descriptors

Objective of the practical session

Learn how to use different landmark detection method, analyze which one is the most suited, and why. Critically analyze the performances of the detectors applied to different exemplar images, and learn how to tune optimally the landmark detection and matching pipeline.

Introduction

Image matching is a fundamental aspect of many problems in computer vision including object and scene recognition. During the development of this practice we are going to analyze the process for extracting distinctive invariant features from images to perform reliable matching among different views of an object or scene. The extracted feature set should be highly distinctive in order to provide robust matching across a substantial range of affine distortions, changes in 3D viewpoint, addition of noise, and changes in illumination. The aim of the lab is to analyze the characteristics of image descriptors and the matching process for detecting a certain pattern. This practical block is expected to be performed in three sessions.

The method

The image descriptors extract from an image a set of significant points in a way that it is consistent with variations of illumination, viewpoint and other viewing conditions The descriptor associates to such points a signature or descriptor which identifies their appearance in a discriminative and compact way, see figure 1a.

The descriptor vectors extracted from different images can be then matched. The matching is often based on a distance between these vectors, e.g. the Euclidean or Mahalanobis distance. A simple way to proceed is to compute the distance between each descriptor in for each point. Then a matched pair from the best score (minimal distance), see figure 1b, is constructed. To increase robustness matches are rejected in relation to nearest neighbor heuristic approaches.

Although the distances approaches described above discard many of the false matches arising from background clutter, we will still may have (false) matches belonging to different objects. Therefore, it is necessary to exclude those points which do not maintain a geometric consistency. For planar scenes this is easy to manage: each possible transformation can be expressed with a homography. A homography (in 3D) is a mapping of a plane to another plane. Mathematically, a homography can be computed by establishing a correspondence between 4 points in each plane. Homographies between pairs of images can be estimated using RANSAC (RANdom SAmple Consensus), see [3]. In case of feature point matching, RANSAC removes outliers in an iterative procedure by randomly selecting a subset of the data and estimating the consensus score based on a classification hypothesis in the remaining set of data.

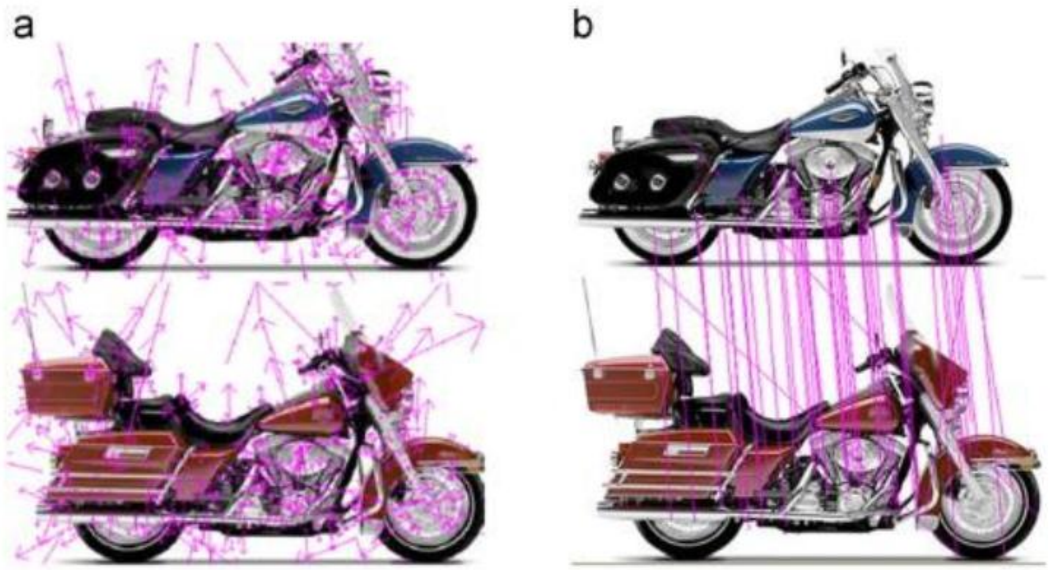


Figure 1: a) 2D detected keypoints and b) matched keypoints.

Practical session 1

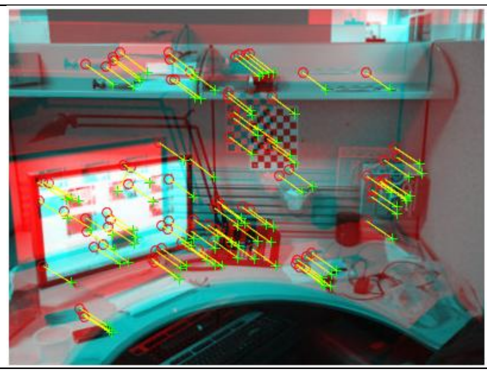
The practical session of the block I consist in coding a landmark detection and aligment between a pair of images.

In order to described the expected result of the pipeline, please have a look to the following two matlab examples:

1) https://es.mathworks.com/help/vision/ug/local-feature-detection-and-extraction.html
openExample('vision/HowToUseLocalFeaturesExample')
this first script illustrates how the image descriptors are used to recover the rotation of an image.

2) https://es.mathworks.com/help/vision/ref/matchfeatures.html
openExample('vision/UseSURFFeaturesToFindCorrespondingPointsExample')
this second script illustrate how the image descriptors are use to recover the alignment between stereoscopic images

The imaging pipeline consist in extracting relevant landmarks using an image descriptor and then matching them using RANSAC.

IMPORTANT, these examples will not be used during the practical session, so it is not important if you don't have the toolbox. They are meant to illustrate exemplar pipelines that the student will reproduce with their own code during the practical session.

The examples are based on concept described into the the three referenced papers at the end of this document. It is recommended to read them before the first session.
During the first session, several 2D image descriptors used in object recognition will be analyzed. Several point detector and image descriptors are described in the paper [4], others are listed in one of the Matlab examples:

Choose a Detection Function Based on Feature Type

Detector	Feature Type	Function	Scale Independent
FAST [1]	Corner	detectFASTFeatures	No
Minimum eigenvalue algorithm[4]	Corner	detectMinEigenFeatures	No
Corner detector [3]	Corner	detectHarrisFeatures	No
SURF [11]	Blob	detectSURFFeatures	Yes
BRISK [6]	Corner	detectBRISKFeatures	Yes
MSER [8]	Region with uniform intensity	detectMSERFeatures	Yes

Note: Detection functions return objects that contain information about the features. The extractHOGFeatures and extractFeatures functions use these objects to create descriptors.

Choose a Descriptor Method

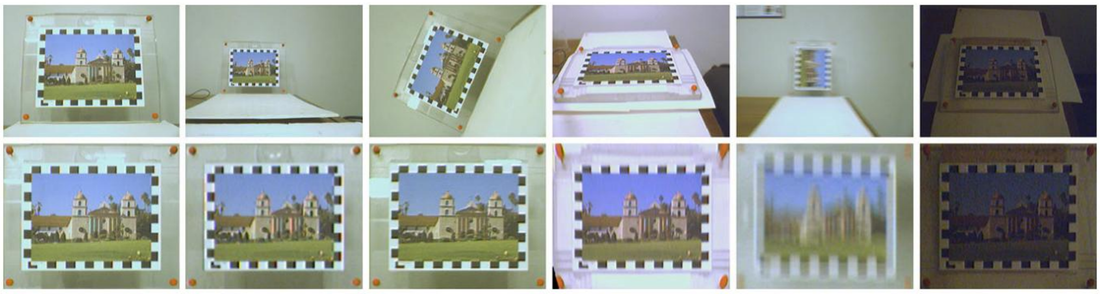
Descriptor	Binary	Function and Method	Invariance		Typical Use	
			Scale	Rotation	Finding Point Correspondences	Classification
HOG	No	extractHOGFeatures(I, ...)	No	No	No	Yes
LBP	No	extractLBPFeatures(I, ...)	No	Yes	No	Yes
SURF	No	extractFeatures(I, 'Method','SURF')	Yes	Yes	Yes	Yes
FREAK	Yes	extractFeatures(I, 'Method','FREAK')	Yes	Yes	Yes	No
BRISK	Yes	extractFeatures(I, 'Method','BRISK')	Yes	Yes	Yes	No
• Block • Simple pixel neighborhood around a keypoint	No	extractFeatures(I, 'Method','Block')	No	No	Yes	Yes

Beside the listed algorithm consider SIFT (Scale-Invariant Feature Transform) as well. The code of this descriptor can be downloaded at:
https://es.mathworks.com/matlabcentral/fileexchange/50319-sift-feature-extreaction
or
https://es.mathworks.com/matlabcentral/fileexchange/43723-sift--scale-invariant-feature-transform--algorithm

session 1 task:

Download several image descriptor codes from the "MATLAB central" web page or from other websites, and test them on sample images provided by Matlab ('cameraman.tif', 'lena', 'viprectification_deskLeft.png',) and others that the student may consider relevant.

For instance the descriptors will be different when applied to the following set of images.



IMPORTANT the image descriptors should not be re-coded, please use the resources available on internet.

Critically compare the different image descriptors to scale, rotation, change in 3D viewpoint, change in illumination invariance and computational efficiency. Which detail of the image are identified by each detector? Which one is the most suited for the selected image? Critically analyze each of them in the PDF report.
Please consider more than one descriptor. Note that it is not necessary to compare all of them. The score of the practical session will depend most of the analysis of the descriptor and not on the amount of descriptors chosen.

Practical session 2

In this second session we will study how the homography between images can be computed, by using the RAMSAC algorithm.
Download the RANSAC code from the page:
https://es.mathworks.com/matlabcentral/fileexchange/30809-ransac-algorithm-with-example-of-finding-homography

As a first step, understand the example. Then use the RANSAC code with the descriptors tested in the previous session. Find the optimal set of parameters of the RANSAC algorithm for each descriptors.

Practical session 3

Reproduce an imaging pipeline similar to the matlab examples provided before. Use the image descriptors studied in the session 1 with the RAMSAC algorithm developed in session2. Compare the results with respect to the matlab examples. Add further examples in order to show the limitations of the algorithm (example panorama stitching)

Deliverable

A **SHORT** report in PDF format containing around five pages must be delivered for the first practical block. The matlab code of the session should be delivered as well. Code everything in a single ZIP file.

- **IN CASE YOU CAN ACCESS TO THE CAMPUS VIRTUAL**

please upload in the corresponding task your file before the dead line

- **IN CASE YOU HAVE NO ACCCESS TO THE CAMPUS VIRTUAL**

CREATE A DROPBOX link with the zip containing the whole deliverable session and send to simone.balocco@ub.edu for evaluation Subject: MAI PR1).
IMPORTANT, do not send the ZIP file directly.

References:

[1] Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints, International Journal of Computer Vision, November 2004.

[2] Bay, H. and Ess, A. and Tuytelaars, T. and Van Gool, L. Speeded-Up Robust Features (SURF), Computer Vision and Image Understanding, June 2008.

[3] Fischler, M.A. and Bolles, R.C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM. June 1981.

[4] S Gauglitz, T Höllerer, M Turk "Evaluation of interest point detectors and feature descriptors for visual tracking" International journal of computer vision, 2011 - Springer