Computer Vision Homework n.1 Report

# PLAYING WITH HOMOGRAPHIES

May 23, 2019

Claudio Curti ID: 4216203
Nicola De Carli ID: 4198668
Alberto Ghiotto ID: 4225586
University of Genoa

# Contents

# Chapter 1

# Correcting the perspective distortion of an image

In the first exercise it was provided the code necessary to create an artificial image of a frontal view of a building, starting from a given image showing the same building from a lateral perspective. To achieve this goal the user has to select 4 points belonging to distorted shapes and 4 points corresponding to virtual edges where it is desired to have the resulting edges. From this correspondences the related homography is estimated using the least squares technique, since it is likely that the resulting equation system has not a solution because of noisy points selected. Eventually, in order to "build" the resulting image, two different implementations were adopted. The first one is the direct mapping from the pixels in the original image to those on the destination image, where it is possible to notice how, by applying the direct mapping, holes appear in the resulting image. This happens because two or more pixels in the source image could correspond to the same pixel in the destination image. To overcome this problem a second implementation is adopted in which the inverse mapping is applied, this method allows to avoid the appearance of the holes. In the provided code the inverse mapping is applied using interpolation which works by doing a weighted average of the values of the surrounding pixels. It was requested to try to apply the inverse mapping without interpolation, in this way the resulting image appear "pixelated". This happens because it is likely that a pixel in the destination image has not a precisely corresponding pixel in the source image and maybe two or more pixels of the destination image correspond to the same pixel in the source image (equivalently as before virtual holes appear), therefore by simply rounding the corresponding pixel in the source image it could happens that discontinuities appear. Furthermore it is possible to notice how, particularly in the church example (fig 1.7), when the assumption of a planar world falls, the part of the image belonging to another plane in the real world result distorted, in this case of the church it is the transept which appears warped.


Figure 1.1: Original building image


Figure 1.2: Direct mapping


Figure 1.3: Inverse mapping


Figure 1.4: Inverse mapping without bilinear interpolation


Figure 1.5: Original cathedral image

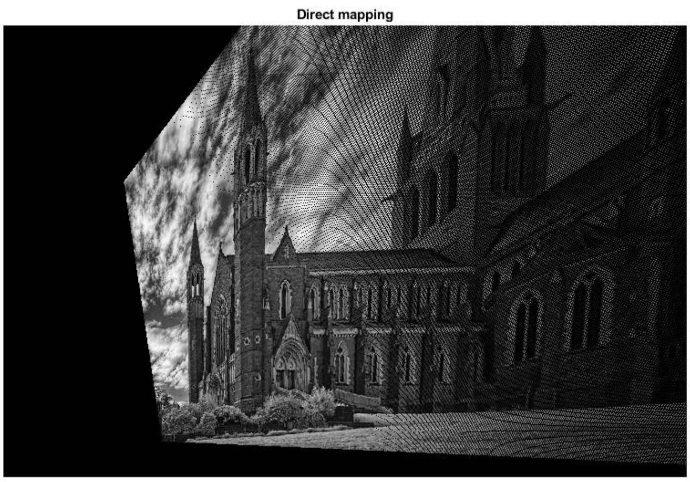


Figure 1.6: Direct mapping

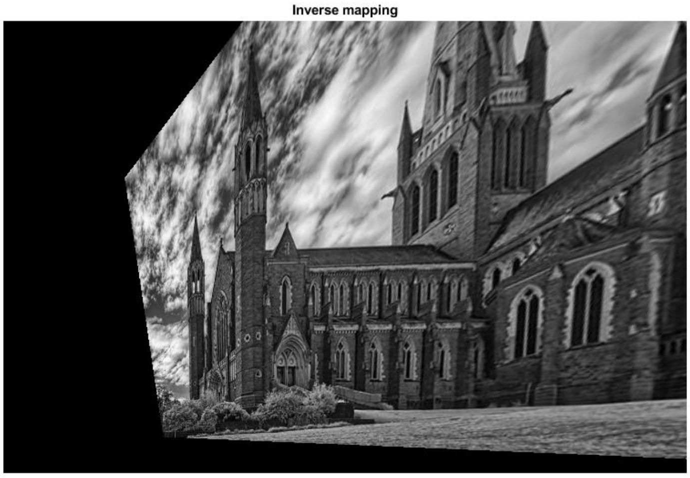


Figure 1.7: Inverse mapping

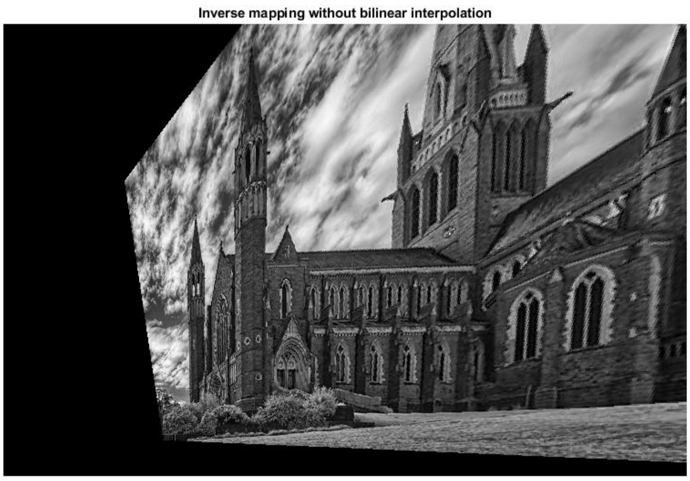


Figure 1.8: Inverse mapping without bilinear interpolation

# Chapter 2

# Estimating the homography between a pair of images

In the second exercise it was provided the code whose aim was to estimate the homography matrix between two images after having selected a certain number of corresponding points. This code works well till the assumption that the object in the real world is planar, holds. Otherwise, if some of the corresponding points are selected from objects which in the real world correspond to different planes, the resulting homography doesn't correspond to the destination image. To overcome this problem a solution is to use the Ransac algorithm, a method to estimate the parameters of a model, starting from a collection of data containing outliers. This algorithm works by iteratively building a model using only a random subset of the data and then computing the number of samples that are distant from this model no more than a certain threshold. If the number of outliers is minor than the previous best result, then the best result is updated and the corresponding model is then set as the best model. In this way, if for example in the image below are selected points both belonging to the notebook surface and to the monster, then the samples belonging to the monster are discarded being treated as outliers and the model is built only using points belonging to the notebook. In order to use this algorithm is important for the samples to be in majority inliners rather than outliers. Moreover, since that in order to compute the homography are required four points, the number of points selected on which to apply the Ransac algorithm, has to be more than four. In the figures below (fig 2.3 and fig 2.4) it is possible to see the results of the Ransac algorithm selecting a set of eight point on the monster.

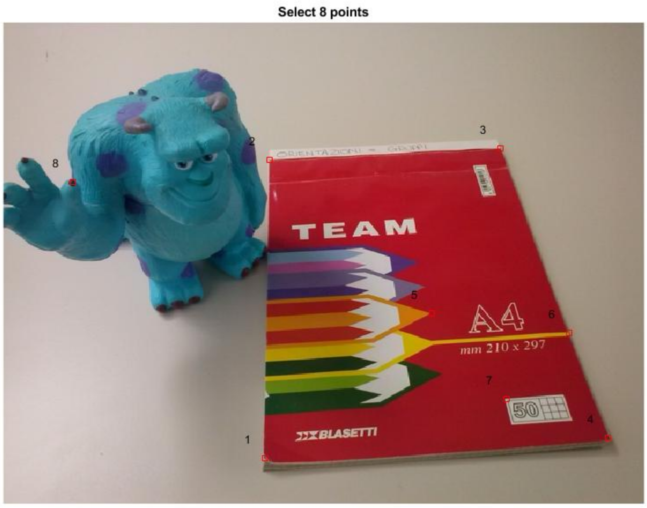


Figure 2.1: Monster original image (first perspective)

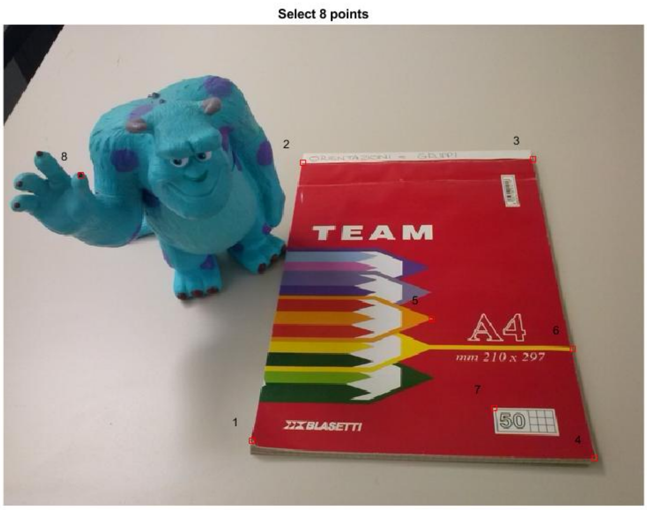


Figure 2.2: Monster original image (second perspective)

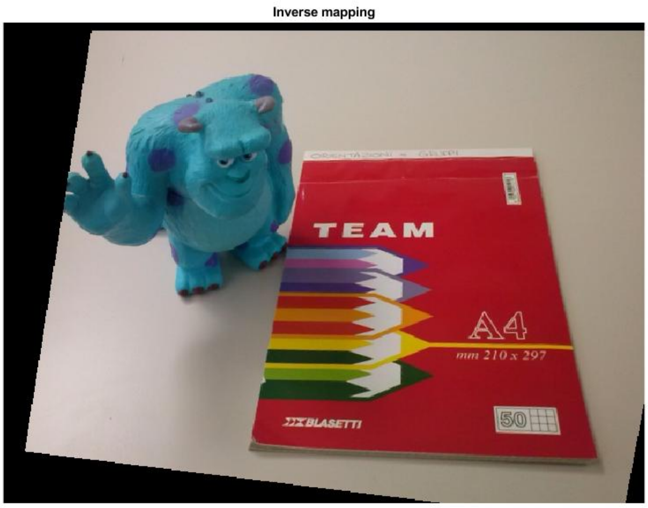


Figure 2.3: Inverse mapping

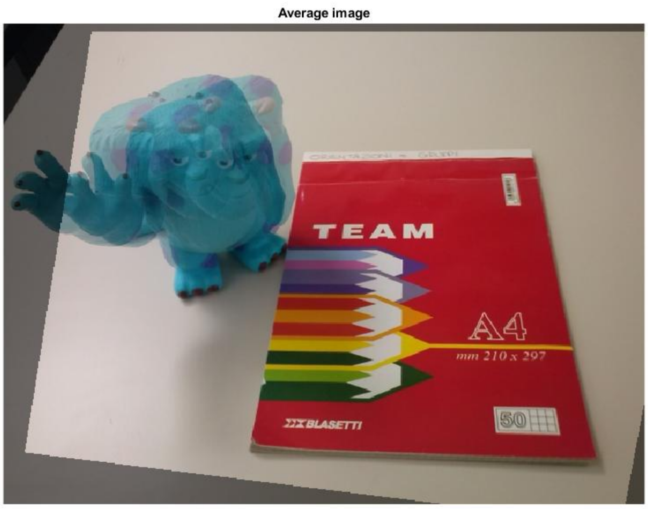


Figure 2.4: Average image

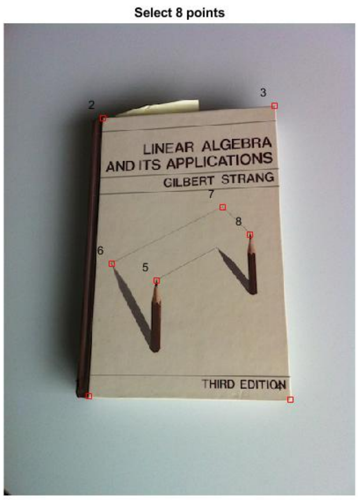


Figure 2.5: Book original image (first perspective)

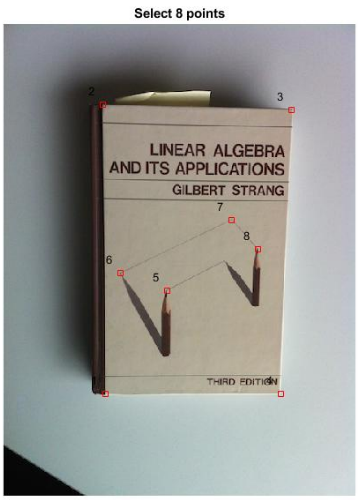


Figure 2.6: Book original image (second perspective)

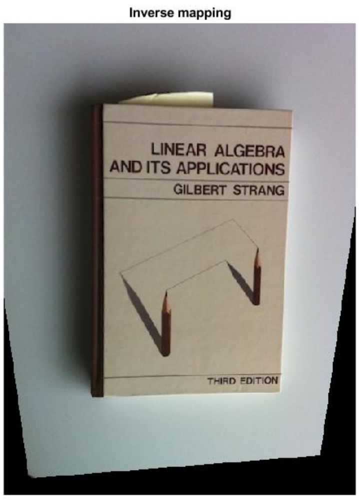


Figure 2.7: Inverse mapping

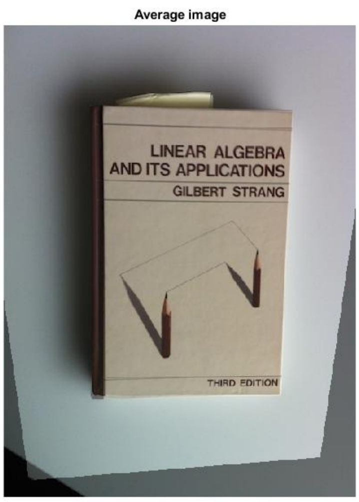


Figure 2.8: Average image

# Chapter 3

# Building a mosaic from a sequence of images

In the third problem the goal is to build a mosaic of different images obtained by rotating a camera, approximating as much as possible a pure rotation. In order to obtain the estimation of the homography of the image with respect to the reference image, corresponding points of each image and a reference central image are selected. Then, to infer the size of the resulting image, the homography of each image with respect to the reference one is applied to each corner, so that at the end the maximum and minimum coordinates are extracted to compute the dimensions. Eventually, the inverse mapping is applied considering all the images used, in this way the images result like glued one over the other in the region of intersection. In order to speed up the selection of the proper points, the set of 7 points for image are selected only on 3 images.

Figure 3.1: Room reconstructed