

University “POLITEHNICA” of Bucharest
Automatic Control and Computers Faculty,
Computer Science and Engineering Department

BACHELOR THESIS

Copyright infringement detection on a
large scale image database

Scientific Advisers:
Prof. Dr. Ing. Nicolae Tapus

Author:
Andrei-Bogdan Parvu

Abstract

Here we can write the abstract of this extraordinary paper

Contents

Abstract	2
1 Design	4
1.1 Analysis of an image	4
1.1.1 Using Harris corner detector	4
1.1.2 Using SIFT keypoints and descriptors	5
1.2 Analysis of a pair of images	5

Chapter 1

Design

1.1 Analysis of an image

1.1.1 Using Harris corner detector

The analysis of an image begins with applying the Harris detector on it. It will compute a given score for each pixel of the image, the higher the score, the greater the chance that pixel represents a corner.

Using these values we will have to determine a subset of pixels that will represent the Harris mask of the image. Experimentally, I have established that all the points which have a value greater than $0.01 * \text{max_image_value}$ are to be part of the subset.

As an example, Figure 1.1 shows a normal image of a woman's face, while Figure 1.2 shows the corresponding pixels that form the corner mask.



Figure 1.1: Sample image

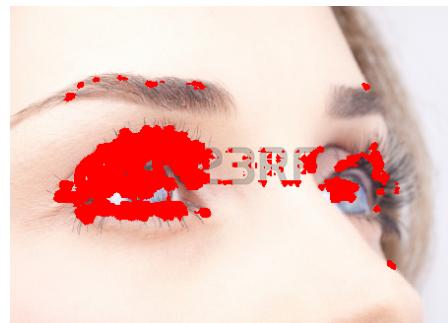


Figure 1.2: Harris corner mask applied

As it can be seen from the image, the corner mask centers around the interest points in the image, the eyes and the eyelashes, while leaving the smooth surfaces (the skin) unmarked.

Another observation is that possible watermarks can be present in the image (as seen above), which, of course, the mask detects (they are center pieces of the image, and the corner algorithm cannot determine that they have no real connection with the image).

Furthermore, in some cases the watermark might contain all the points in the mask (as seen in Figure 1.3). To avoid such a behavior, the image is split into 9 sub images (three rows and three columns of identical dimensions) and the Harris mask is applied to each of these.



Figure 1.3: Single image



Figure 1.4: 9 subimages

This determines the corner mask to also contain pixels from the center of the image (some of which, unfortunately, are also a watermark).

1.1.2 Using SIFT keypoints and descriptors

As we have seen, in ceva, the SIFT algorithm computes the keypoints of an image, and then computes descriptors for these keypoints. Due to the high similarity nature of our problem, we do not want to compute the keypoints for the entire image, but filter them based on the corner mask determined in the previous section (as observed in Figure 1.5 and Figure 1.6)

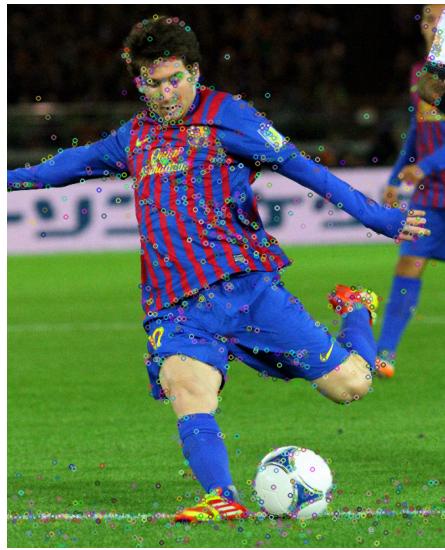


Figure 1.5: SIFT keypoints for the entire image



Figure 1.6: SIFT keypoints for corner mask

The SIFT descriptors are then computed for each keypoint located in the corner mask and this will be the information stored for a certain image.

1.2 Analysis of a pair of images