Detection and Recognition of RGB-D and 3D objects

Miquel Perelló, E-mail: miquel.perello.nieto@est.fib.upc.edu

Abstract—In this work I analysed newest techniques in object detection in 2D and 3D spaces. For 2D images, analysis are performed with openCV libraries. In the other case, it is theoretically explained.

Keywords—Object Recognition, SIFT, SURF, SHOT, CSHOT.

INTRODUCTION

THIS work is divided in the different techniques I analysed. These are one per section in this order Scale-Invariant Feature Transform (SIFT), Speeded Up Robust Features (SURF), Signatures of Histograms for Local Surface (SHOT) and finally Color Signature of Histograms for Local Surface (CSHOT).

There is also in the middle one section with a comparison between SIFT and SURF techniques.

1 SIFT

This technique is intended to find keypoints invariant to image scale and rotation. It also performs good results with affine distortions, change in 3D viewpoints, addition of noise and change in illumination.

1.1 Octave layers

For searching SIFT points is needed to perform a Differnce of Gaussian (DoG) between different scales. For that reason, the initial image is scaled several times (each of these is a octave scale). Inside each octave is performed several Laplacian of Gaussians. And for each pair it is performed the DoG mentioned before.

The number of octave layers determines the number of intervals to create for each octave. Concretely, the number of blurred images is $s + nOctaveLayers$, where $s \mid k = 2^{1/s}$.

I have performed a test execution, trying different number of layers and repeating every execution for twenty similar images. The results shows that increasing number of layers it detects more points (figure 1). But this points are mainly the same points in different scales and may be are not so different. Furthermore, the time it spends to compute this points increases linearly (figure 2), but it does not give much more information about the picture.

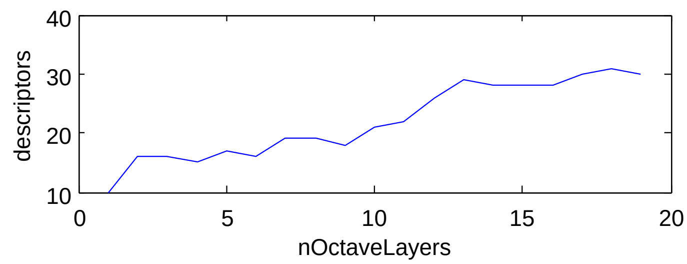

Fig. 1. SIFT octaves respect number of descriptors

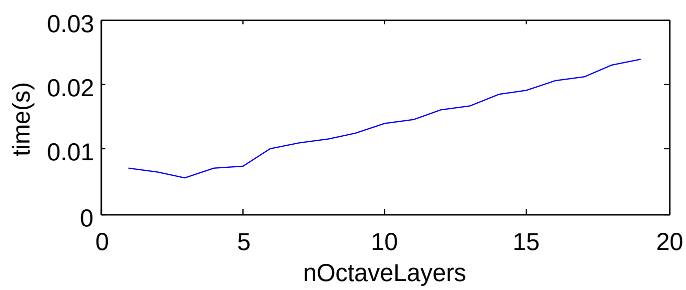

Fig. 2. SIFT octaves respect time

For that reason [1, Lowe et al] selects 3 layers, it is a trade-off between computation time and number of descriptors.

1.2 Contrast threshold

This is a threshold that controls which descriptors discard directly. It removes these points

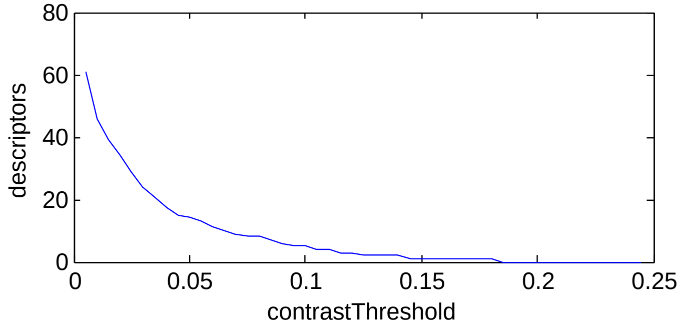

Fig. 3. SIFT contrast threshold respect number of descriptors

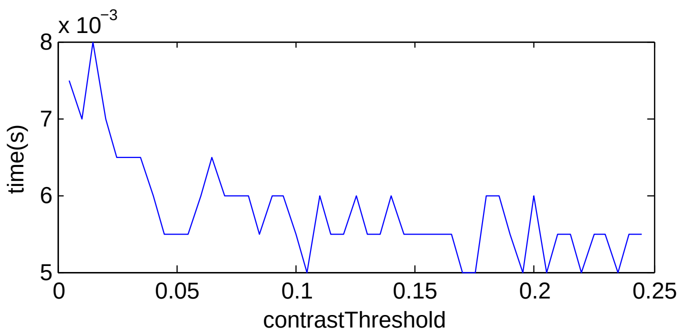

Fig. 4. SIFT contrast threshold respect time

if its contrast is beside this value. Incrementing the contrast threshold the number of descriptors is reduced considerably (figure 3). One threshold large enough to select sufficient points is 0.04.

Execution time is also reduced, but the reduction is very low (figure 4).

1.3 Edge threshold

In that case this threshold imposes some restrictions in the degrees of the descriptors. For example, we can discard all descriptors which angles are below 30 degrees. The result is an increment of descriptors with bigger values of threshold (figure 5).

Computation time remains less or more in the same range (figure 6).

1.4 Sigma value

Sigma value controls the standard deviation of gaussian filters. Depending on the image quality, it is recommended to use small values of sigma. This is because loss of information in each filter application.

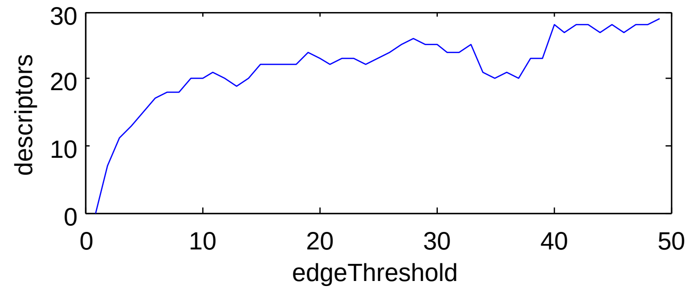

Fig. 5. SIFT edge threshold respect number of descriptors

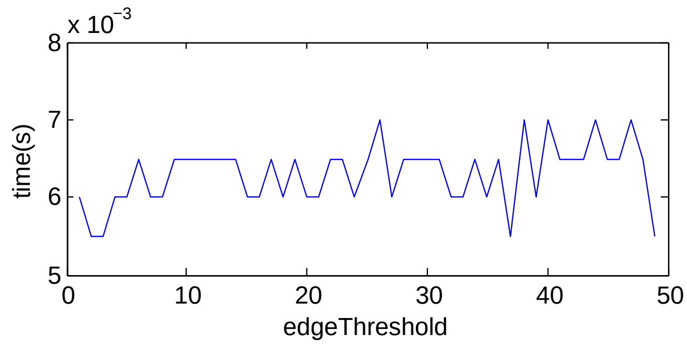

Fig. 6. SIFT edge threshold respect time

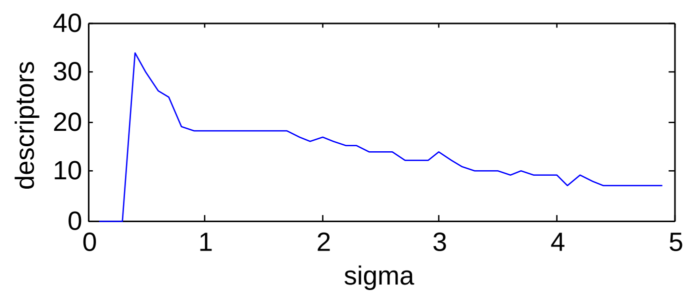

Fig. 7. SIFT sigma respect number of descriptors

In this case we can see the number of descriptors decreasing when incrementing sigma value (figure 7), while computation time increases linearly (figure 8).

Recommended value in [1, Lowe et al] is 1.6.

2 SURF

This technique by Bay et al. [2] is a novel scale- and rotation-invariant interest point detector and descriptor.

2.1 Octaves

In SURF algorithm uses Gaussian smooths applied at initial image. This Gaussian is applied

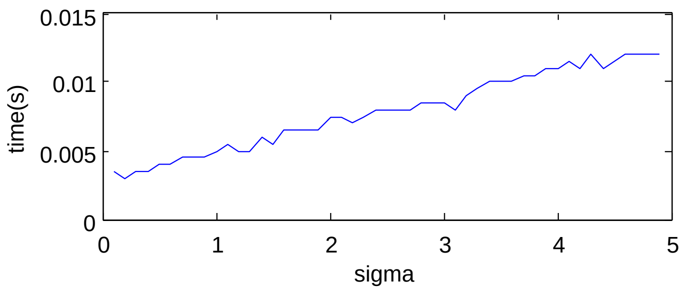

Fig. 8. SIFT sigma respect time

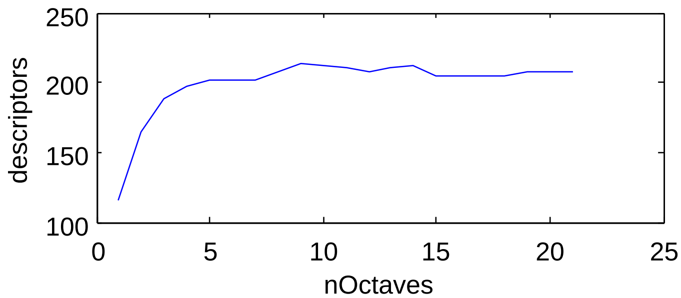

Fig. 9. SURF octaves respect number of descriptors

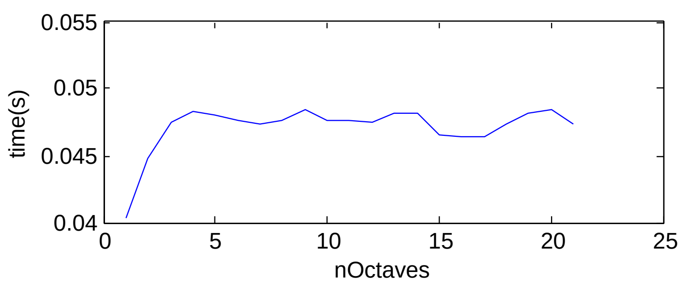

Fig. 10. SURF octaves respect time

several times and usually named pyramid. Results shows the number of descriptors detected remains stable at certain point (figure 9). Same happens with computation time (figure 10).

Four is a good value for that parameter. It have also layers in each octave, these results are not shown here, but usually two layers are selected.

2.2 Hessian threshold

This parameter determines which keypoints are discarded by his Hessian matrix. SURF detector uses a Fast-Hessian detector to choose the best keypoints. In that case larger Hessian value is, less points are selected (see figure 11). Computation time also decreases (see figure 12).

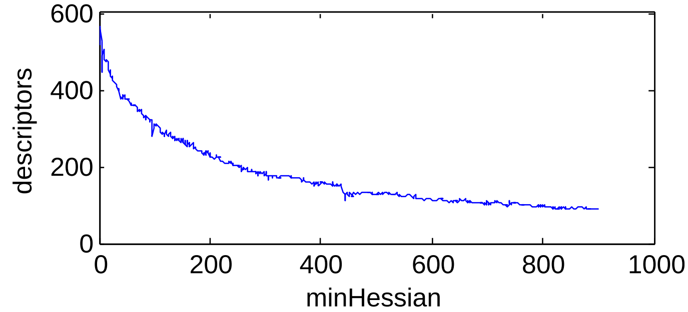

Fig. 11. SURF hessian threshold respect number of descriptors

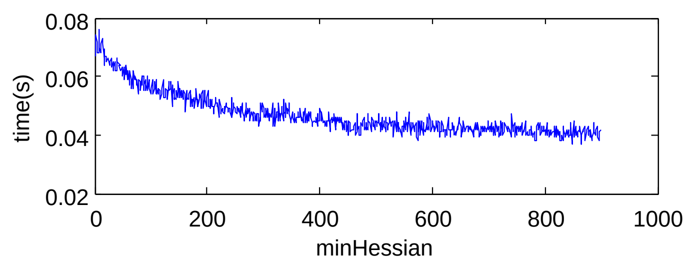

Fig. 12. SURF hessian threshold respect time

2.3 Upright flag

It computes orientation of features. It is relevant if you want to retrieve rotated objects. In that case, every keypoint is rotated to zero respect the biggest value in the histogram.

In my execution examples we can not see difference between both. But number of descriptors in these images was not so huge. It is expected to increase computation time for normalising keypoints.

upright	Hessian	Lay.	Feat.	Descr.	time
false	300	4	128	102	41.5
true	300	4	128	107	41.5

3 SIFT VS SURF

These two descriptors have been compared with the webcam and localising a folder. If we compare the computational efficiency of SIFT respect SURF without upright, there is a big difference between both. SIFT spends about 700ms to compute the descriptors and match, while SURF is about 200ms. If the upright option is turned on, the difference drops to half. In that case SURF spends about 450ms.

About scale and illumination, both seems to behave very similar. On the other side, it seems that (in my case) rotations are better detected by SIFT. On the contrary, 3D view points seems to perform better with SURF descriptors.

4 RANSAC

This method allows to detect objects in a very fast way. It relies in the fact that some of our keypoints are enough to detect in a picture. It make several random chooses in the picture, and try to fit in searched object. With this simple idea, it performs very well and in a cheap computational cost.

5 SHOT

This is a new technique to detect objects in a 3D space. It is based in some way in SIFT techniques. Histograms have demonstrated in SIFT and other techniques very good results in object detection. Furthermore, these attributes are very fast to compute.

In this approach instead of computing a Histogram of Gradient (HOG) on the black & white image, it computes the histogram of gradients on several points in the surface of the objects. The idea is very similar to SIFT, but in that case features remains in three dimensions.

One of the problems is that only match the objects surface. This is fine for some scenarios, but there are a lot of environments where colour is also important.

6 CSHOT

This technique improves SHOT's accuracy adding object texture features in description. It relies in two histograms, one of surface shape and other for texture/colour. Tombari et al. [4] affirms this technique outperforms SHOT results.

7 Conclusion

We have seen a brief summary of newest techniques in object detection. Focusing on computational efficiency and variations in illumination, scale, rotation and 3D viewpoint.

References

[1] Lowe, David G. ”Distinctive image features from scale-invariant keypoints.” International journal of computer vision 60.2 (2004): 91-110.

[2] Bay, Herbert, Tinne Tuytelaars, and Luc Van Gool. ”Surf: Speeded up robust features.” Computer Vision–ECCV 2006. Springer Berlin Heidelberg, 2006. 404-417.

[3] Tombari, Federico, Samuele Salti, and Luigi Di Stefano. ”Unique signatures of histograms for local surface description.” Computer Vision–ECCV 2010. Springer Berlin Heidelberg, 2010. 356-369.

[4] Tombari, Federico, Samuele Salti, and Luigi Di Stefano. ”A combined texture-shape descriptor for enhanced 3d feature matching.” Image Processing (ICIP), 2011 18th IEEE International Conference on. IEEE, 2011.