Electronic Letters on Computer Vision and Image Analysis 12(2):88-97; 2013

Image based Monument Recognition using Graph based Visual Saliency

Grigorios E. Kalliatakis* and Georgios A. Triantafyllidis**

** Applied Informatics and Multimedia Dept., Technological Educational Institute of Crete, Heraklion, Greece*
*** Medialogy Section, Aalborg University Copenhagen, Denmark*
+

Received 5th Nov 1012; accepted 21st Dec 2012

Abstract

This article presents an image-based application aiming at simple image classification of well-known monuments in the area of Heraklion, Crete, Greece. This classification takes place by utilizing Graph Based Visual Saliency (GBVS) and employing Scale Invariant Feature Transform (SIFT) or Speeded Up Robust Features (SURF). For this purpose, images taken at various places of interest are being compared to an existing database containing images of these places at different angles and zoom. The time required for the matching progress in such application is an important element. To this goal, the images have been previously processed according to the Graph Based Visual Saliency model in order to keep either SIFT or SURF features corresponding to the actual monuments while the background "noise" is minimized. The application is then able to classify these images, helping the user to better understand what he/she sees and in which area the image has been taken. Experiments are performed to verify that the proposed approach improves the time needed for the classification without affecting the correctness of the results.

Key Words: SIFT, SURF, Graph Based Visual Saliency, Image classification

1. Introduction

The matching of images in order to establish a measure of their similarity is a major problem in many computer vision tasks. Object recognition, image registration and building panoramas represent just a small sample of possible applications [1]. Moreover with the popularization of digital cameras and mobile phones, more individuals are able to take pictures that can be shared in the Internet [2]. A crucial task would be to automatically classify photographs taken by an individual at different places, in order to allow richer user interaction and support new exciting applications. This approach is characterized as classifying a number of images into different categories, where each category is composed of images that have a similar content, in terms of representing a monument.

The problem of image classification remains especially challenging when considering outdoor images, which originate from a diversity of environments. Our goal in this paper is to effectively classify images of monuments by consuming as least as possible computational time compared to the traditional approaches. As shown in Figure 1, we employ a specific pre-processing scheme based on Graph Based Visual Saliency [3]

Correspondence to: <gt@create.aau.dk>

Recommended for acceptance by <Angel Sappa>
ELCVIA ISSN: 1577-5097
Published by Computer Vision Center / Universitat Autonoma de Barcelona, Barcelona, Spain

in order to minimize the effect of features not belonging to the monument, which delay and even mislead the classification result. The experimental results show a significant improvement of the required computational time without affecting the classification results compared to the SIFT or SURF based classification using the original images without the proposed pre-processing stage.

The rest of the paper is organized as follows: Section 2 refers to SIFT algorithm use and implementation, while section 3 refers to SURF algorithm. In Section 4, we briefly introduce the meaning and the use of the Graph Based Visual Saliency. Section 5 elaborates the proposed method while in Section 6 both the experimental and the performance results are given. Finally, in section 7 we provide some thoughts on future works for this application.

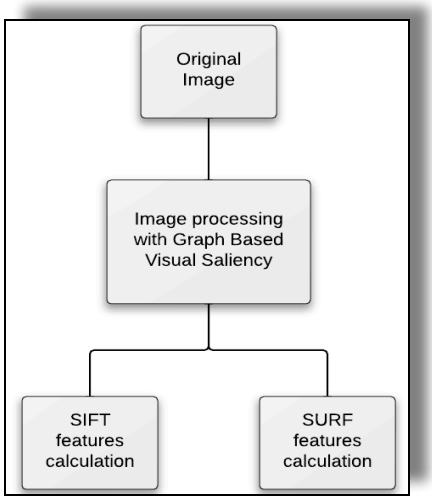

Figure 1 Proposed Scheme

2. Scale Invariant Feature Transform (SIFT) algorithm

The Scale Invariant Feature Transofrm (SIFT) algorithm, developed by David G. Lowe [4],[5],[6] is an algorithm for image features generation which act as descriptors of local image patches. These features are reasonable invariant to scalling, translation, rotation and partially invariant to illumination changes and affine projection. Calculation of SIFT image features is performed through the four steps briefly described in the following:

1. **Scale – space extrema detection:** The first stage of computation searches over all image locations and scales. It is implemented efficiently by using a difference – of- Gaussian (DOG pyramid) function to identify potential interest points that are invariant to scale and orientation. To build the DOG pyramid the input image is convolved iteratively with a Gaussian kernel of $\sigma = 1.6$. The last convolded image is down-sampled by 2, in each image direction and the convolving process is repeated. All the collection of images of the same size build together the so-called Gaussian pyramid. The local extrema (maxima or minima) of DOG function are detected by comparing each pixel with its 26 neighbours.

2. **Keypoint Localization:** At each candidate location, a detailed model is fit to determine location and scale. Keypoints are selected based on measueres of their stability. The detected local extrema need to be exactly localised by fitting a 3D quadratic function to the scale-space local sample point. Local extrema with low contrast and such that correspond to edges are discarded due to their sensitivity to noise.

3. **Orientation assignment:** One or more orientations are assigned to each keypoint location based on local image gradient directions. All future operations are performed on image data that has been trasnformed relative to the assigned orientaion, scale, and location for each feature, thereby providing invariance to these transformations. There will be multiple keypoints created at the same location but with different orientations.

4. **Keypoint descriptor:** The local image gradients are measured at the selected scale in the region around each keypoint are divided into 4x4 boxes. Then for each box an 8 bins orientaion histogram is

established. These are transformed into a representation, a 128 dimensional vector (SIFT-descriptor), that allows for significant levels of local shape disortion and change in illumination.

Figure 2 illustrates an eaxmple of SIFT keypoints descriptors belonging to one of the monuments of the database.

Figure 2 SIFT Keypoint Detection

The process for image matching is as following: First the distance of all feature points between two images is calculated. Then the ratio of the nearest neighboring distance (NN) to the second neighboring distance (SN) is calculated. When R=NN/SN is less than a constant, the feature points between the two images are matching. Lowe suggests R to be equal to 0.6.

3. Speeded-Up Robust Features (SURF) algorithm

Speeded-Up Robust Features is a fast and robust algorithm for local, similarity invariant image representation and comparison. The SURF framework is based on the PhD thesis of H.Bay [7]. Similarly to the SIFT approach, SURF selects interest points of an image from the salient features of its linear scale-space, and then builds local features based on the image gradient distribution. The main interest of this approach lies in its fast computation of approximate differential operators in the scale-space, enabling real-time applications such as the proposed one in this article. The SURF algorithm is composed of three consecutive steps:

1. **Interest point detection:** The local maxima of the Hessian determinant operator applied to the scale-space are computed in order to select interest point candidates. These candidates are then validated if the response is above a given threshold.

2. **Interest point description:** The purpose of this step is to build the actual descriptor that is invariant to view-point changes of the local neighborhood of the point of interest. Making use of a spatial localization grid, a 64-dimensional descriptor is built, corresponding to a local histogram of the Haar wavelet responses.

3. **Feature matching:** The final step matches the descriptors of both images. Comparisons are performed by computing Euclidean distance between all potential matching pairs. A matching criterion based on nearest-neighbor ratio is then used to reduce mismatches. After these filters, one can be sure that the remaining matches real and correspond to the same scene seen from different viewpoints.

The dimension of the descriptor has a direct impact on the time this takes, and less dimensions are desirable for fast interest point matching. However, lower dimensional feature vectors are in general less distinctive than their high-dimensional counterparts. Figure 3 shows an example of SURF keypoints descriptors extracted from one of the monuments of the database.

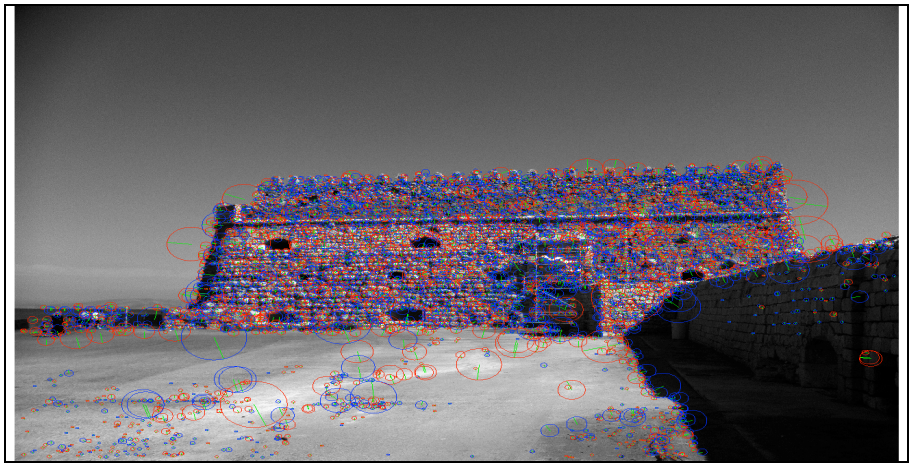

Figure 3 SURF Keypoint Detection

4. Graph Based Visual Saliency (GBVS)

Saliency should be defined as the discriminativeness of features. Saliency maps contain information about where interesting information can be found in the image. These areas correspond to features considered as rare or informative, depending on the definition of saliency. High saliency regions correspond to objects or places they are most likely to be found, while lower saliency is associated to background.

In this context, a distributed graph-based solution called Graph-Based Visual Saliency (GBVS) is proposed in [3]. The main idea is to find saliency values at each location which depend on the entire image plane. This is different than most other modern approaches which rely on local information.

So the method itself consists of two steps: first forming activation maps on certain feature channels, and then normalizing them in a way, which highlights conspicuity and admits combination with other maps. The model is simple and biologically plausible insofar as it is naturally parallelized. Given an image, we wish to ultimately highlight a handful of 'significant' locations where the image is 'informative' according to some criterion. This process is conditioned on first computing feature maps e.g. by linear filtering followed by some elementary nonlinearity. Interpreting this technique we can conclude that the algorithm allows the mass of the prominent points away from the boundaries of the object with a non-trivial way that cannot be achieved only by the smoothing.

Figure 4 shows an original image, along with the corresponding GBVS map. The third picture of Figure 4 is the original image with some blanked areas derived from the GBVS map, which cover the 30% of the whole image. This percentage selection of 30% was made heuristically, as the most appropriate number in terms of reducing background scenery without affecting the dominant object of each image. This decision was taken considering the nature of the original images we had to process. In other situations that percentage could easily have been adjusted as desired by the user.

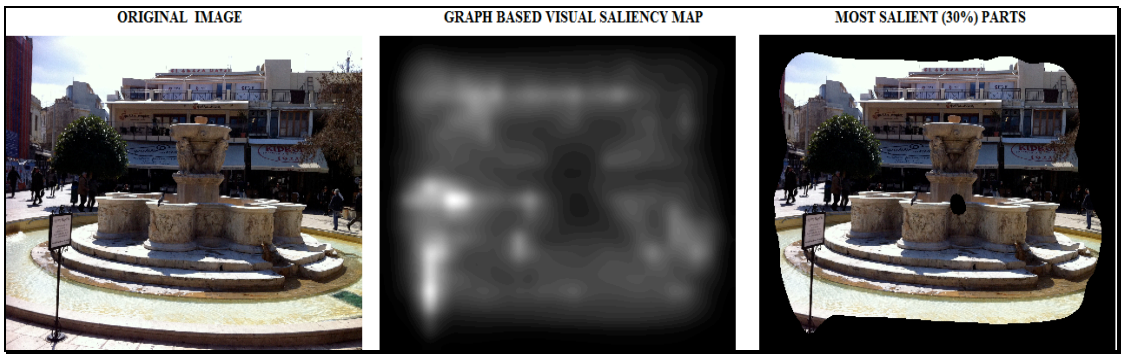

Figure 4 Example of Graph Based Visual Saliency Map

5. Proposed method

The proposed method deals with the problem of time inefficiency of many image classification applications. The problem is actually caused by the presence of many SIFT/SURF keypoints that do not belong to the object we want to classify. These keypoints delay and even may mislead the classification result. Based on this fact, it is obvious that if we could have the 'significant' locations of an image (where the image is 'informative'), this would be a great advantage for the classification task. In this context, we propose the use of a pre-processing stage that employs the GBVS algorithm in order to minimize the "noise" of the monument's irrelevant information and keep only the SIFT features that will help faster and more accurate classification. The application is then able to classify the images fast and correct.

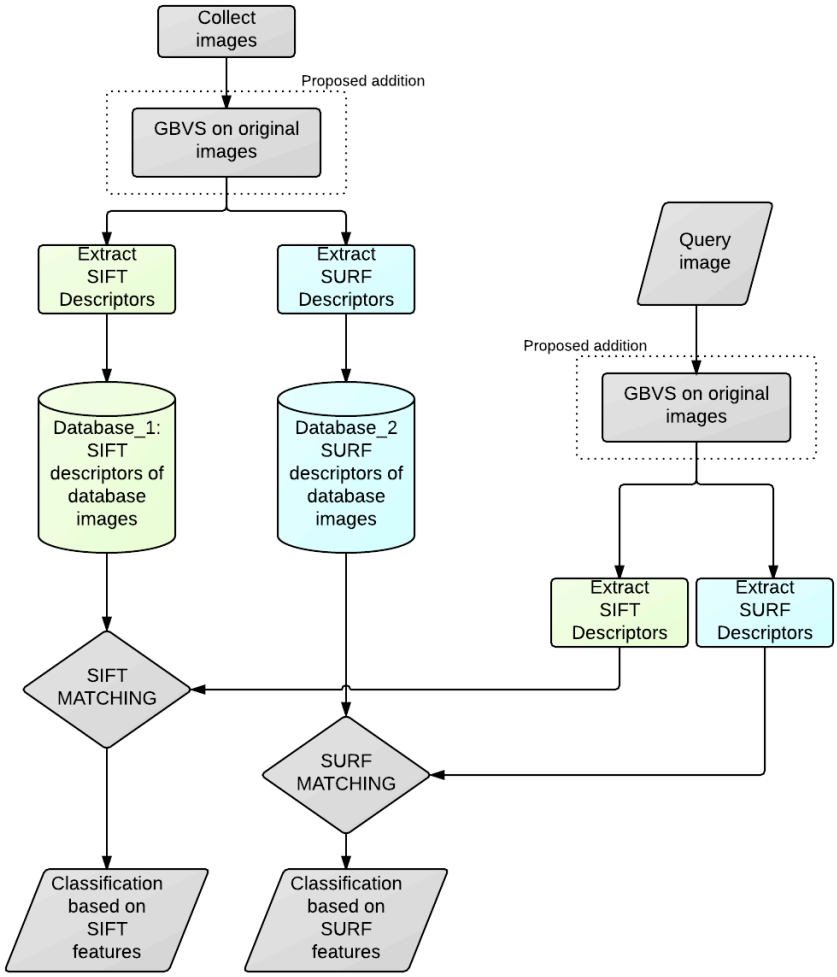

Figure 5 Experiment overview

The process (see Figure 5) followed towards our experimental results initially contain the creation of the database featuring 4 classes of specific monuments in Heraklion, Crete (church, fortress, fountain, loggia - see Figure 6) in which the query image can be classified. To this goal, we collect 5 photos of each of these classes. Then, we apply the GBVS on original images in order to have them blanked if some areas where lower saliency they are associated to background. By doing that and during the SIFT and SURF descriptors calculation, we will save time by discarding the unnecessary keypoints associated to background (see Figure 7).

For the comparison, we actually build two SIFT and two SURF databases, one with the features/descriptors of the original images and the other with the features/descriptors of the pre-processed

images using the GBVS. Also, the query image can be either used as it is or pre-processed by the GBVS, for the following calculation of the SIFT, SURF features.

Figure 6 The four monuments in Heraklion, Crete, considered in our application

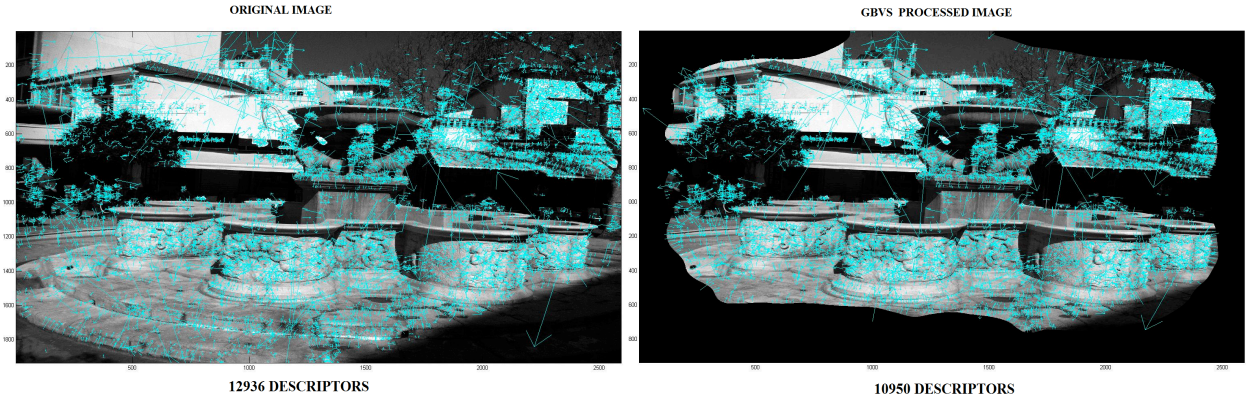

Figure 7 Rejecting redundant keypoints

Having a query image to be classified in one of the aforementioned classes, we perform a simple matching with both SIFT and SURF databases in order to test the needed time. Our Classification Method is a simple one, based on aggregating the total number of matches of each class of the database which being tested. This can be achieved by comparing one query image object and one database image object at a time for every class in our database, containing either original images or pre-processed. After that, each class is marked with one total number of matches compared to the query image. The class with the higher value in this field is considered to be the target class for the query image.

6. Results

We must refer to some general considerations regarding our experiment. Firstly all the photographs that took place in the experimental results were taken with an iPhone 4 camera shooting at 5MP analysis. Also we must note that the time needed for the query image to be pre-processed by GBVS is already included in the results of Table 1 & Table 2.

We calculate the computation time and the classification results, for each scene using the SIFT algorithm in Table 1 and using the SURF algorithm in Table 2.

Table 1 Classification & time saved using SIFT

Query Image	ORIGINAL	SALIENCY ON DB IMAGES ONLY	TIME SAVED	CORRECT CLASS	SALIENCY ON BOTH DB & QUERY IMAGES	TIME SAVED	CORRECT CLASS
Church_1	147 sec 1298 matches	138 sec 1191 matches	**9 sec**	√	128 sec 1167 matches	**19 sec**	√
Church_2	266 sec 30 matches	247 sec 47 matches	**19 sec**	√	203 sec 51 matches	**63 sec**	√
Church_3	237 sec 81 matches	210 sec 65 matches	**27 sec**	√	191 sec 75 matches	**46 sec**	√
Fortress_1	115 sec 331 matches	107 sec 327 matches	**8 sec**	√	109 sec 352 matches	**6 sec**	√
Fortress_2	146 sec 131 matches	143 sec 135 matches	**3 sec**	√	128 sec 150 matches	**18 sec**	√
Fortress_3	128 sec 62 matches	110 sec 53 matches	**18 sec**	√	65 sec 58 matches	**63 sec**	√
Fountain_1	228 sec 1027 matches	217 sec 957 matches	**11 sec**	√	211 sec 968 matches	**17 sec**	√
Fountain_2	258 sec 199 matches	248 sec 187 matches	**10 sec**	√	209 sec 190 matches	**49 sec**	√
Fountain_3	336 sec 274 matches	328 sec 225 matches	**8 sec**	√	275 sec 181 matches	**61 sec**	√
Loggia_1	269 sec 155 matches	209 sec 131 matches	**60 sec**	√	145 sec 146 matches	**124 sec**	√
Loggia_2	220 sec 289 matches	208 sec 259 matches	**12 sec**	√	194 sec 233 matches	**26 sec**	√
Loggia_3	215 sec 184 matches	202 sec 151 matches	**13 sec**	√	157 sec 153 matches	**58 sec**	√

Table 2 Classification & time saved using SURF

Query Image	ORIGINAL	SALIENCY ON DB IMAGES ONLY	TIME SAVED	CORRECT CLASS	SALIENCY ON BOTH DB & QUERY IMAGES	TIME SAVED	CORRECT CLASS
Church_1	103 sec 1394 matches	97 sec 1250 matches	**6 sec**	√	82 sec 1188 matches	**20 sec**	√
Church_2	248 sec 205 matches	233 sec 287 matches	**15 sec**	√	182 sec 183 matches	**65 sec**	√
Church_3	206 sec 415 matches	193 sec 414 matches	**13 sec**	√	172 sec 409 matches	**33 sec**	√
Fortress_1	213 sec 1891 matches	197 sec 1921 matches	**16 sec**	√	193 sec 1904 matches	**19 sec**	√
Fortress_2	273 sec 1634 matches	259 sec 1620 matches	**14 sec**	√	244 sec 1599 matches	**28 sec**	√
Fortress_3	212 sec 361 matches	176 sec 370 matches	**36 sec**	√	111 sec 218 matches	**100 sec**	√
Fountain_1	268 sec 1757 matches	221 sec 1626 matches	**47 sec**	√	211 sec 1603 matches	**56 sec**	√
Fountain_2	300 sec 424 matches	238 sec 421 matches	**62 sec**	√	211 sec 408 matches	**88 sec**	√
Fountain_3	348 sec 546 matches	299 sec 500 matches	**49 sec**	√	260 sec 400 matches	**87 sec**	√
Loggia_1	166 sec 167 matches	145 sec 157 matches	**21 sec**	√	91 sec 146 matches	**74 sec**	√
Loggia_2	197 sec 326 matches	167 sec 305 matches	**30 sec**	√	155 sec 288 matches	**41 sec**	√
Loggia_3	178 sec 297 matches	145 sec 270 matches	**33 sec**	√	109 sec 195 matches	**68 sec**	√

In order to measure the performance of our experimental results, we employ the Root-Mean-Square Deviation (RMSD). This is a commonly used measure of the differences between values predicted by a model or an estimator or the values actually observed by an experiment (such as in our case):

$$\mathrm{RMSD}(\theta_1,\theta_2) = \sqrt{\mathrm{MSE}(\theta_1,\theta_2)} = \sqrt{\mathrm{E}((\theta_1-\theta_2)^2)} = \sqrt{\frac{\sum_{i=1}^{n}(x_{1,i}-x_{2,i})^2}{n}}.$$

We are measuring both matching and time performance in our application.

- Regarding the time, we set θ_1 the time elapsed on matching when query and database images are both pre-processed with saliency, while the original and the only saliency-processed database images are noted as θ_2.
- Regarding the matches, we set as θ_1 the matches found on the original query and database images. Matches on saliency-processed database images or matches on both saliency-processed database images and query images, are noted as θ_2.

In Table 3 the root-mean-square deviation for time and matches are shown for both of the algorithms (SIFT and SURF) we employed in our experiment. The results show the effectiveness of the proposed pre-processing GBVS approach in image classification compared to the traditional approach.

ALGORITHM	***TIME*** *Root Mean Square Error BOTH SALIENCY vs ORIGINAL*	***TIME*** *Root Mean Square Error BOTH SALIENCY vs DB_SALIENCY*	***MATCHES*** *Root Mean Square Error ORIGINAL vs DB_SALIENCY*	***MATCHES*** *Root Mean Square Error ORIGINAL vs BOTH SALIENCY*
SIFT	**62**	**33**	**64**	**96**
SURF	**55**	**35**	**42**	**53**

Table 3 Matches & time RMSD

7. Conclusion and future work

This paper presents a method derived from the problem of automatic outdoor image classification. The proposed approach computes the similarity between two images to ultimately classify the query image. It is concluded that employing the Graph Based Visual Saliency as a pre-processing stage, reduces the time needed in a matching process based on SIFT and SURF keypoints. It also improved the matching performance. In this context we developed an application for simple image classification of well-known monuments in the geographic area of Heraklion, Crete, Greece.

Future work will be rooted on improving the total time required for the classification method. This can be done by not simple comparing the query image to each one of the database images, but employing more sophisticated methods for classification (eg BoF, codebooks, etc). Also, a hierarchy of the SIFT/SURF keypoints based on the GBVS will be investigated.

References

[1] Faraj Alhwarin, Chao Wang, Danijela Risti -Durrant, Axel Gräser: ”Improved SIFT-Features Matching for Object Recognition”, BCS International Aca demic Conference 2008 – Visions of Computer Science

[2] Oliver van Kaick and Greg Mori: "Automatic Classification of Outdoor Images by Region Matching", Third Canadian Conference on Computer and Robot Vision (CRV 2006), 7-9 June 2006, Quebec City, Canada 2006

[3]Jonathan Harel, Christof Koch, and Pietro Perona, "Graph-Based Visual Saliency", Advances in Neural Information Processing Systems 19, pages 545-552, 2007

[4] David G.Lowe. : Object recognition from local scale-invariant features, International Conference on Computer Vision, Corfu, Greece (September 1999)

[5] David G. Lowe. : Local feature view clustering for 3D object recognition, IEEE Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii (December 2001),

[6] David G. Lowe. : Distinctive image features from scale-invariant key-points, International Journal of Computer Vision, 60, 2 (2004)

[7] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool: "SURF: Speeded Up Robust Features". Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346—359, 2008

[8] Root-mean-square deviation on http://en.wikipedia.org/wiki/Root-mean-square_deviation