Evaluation of Features Detectors and Descriptors based on 3D objects

I. M. Anonymous M. Y. Coauthor

Abstract

We explore the performance of a number of popular feature detectors and descriptors in matching 3D object features across viewpoints and lighting conditions. To this end we design a method, based on intersecting epipolar constraints, for providing ground truth correspondence. We also collect a database of 100 *objects viewed from* 144 *calibrated viewpoints under three different lighting conditions. We find that the combination of Harris feature finder and SIFT features is most robust to viewpoint change, while combining Harris with steerable filters is the best for lighting changes. We also find that no detector-descriptor combination performs well with viewpoint changes of more than 25-30°.*

1 Introduction

Detecting and matching specific visual features across different images has been shown to be useful for a diverse set of visual tasks including stereoscopic vision [1, 2], vision-based simultaneous localization and mapping (SLAM) for autonomous vehicles [3], mosaicking images [4] and recognizing individual objects [5, 6]. This operation typically involves three distinct steps. First a 'feature detector', also called 'feature finder' or 'interest operator', identifies a set of image locations presenting rich visual information and whose spatial location is well defined. The spatial extent or 'scale' of the feature may also be identified in this first step. The second step is 'description': a vector characterizing local texture is computed from the image near the nominal location of the feature. 'Matching' is the third step: a given feature is associated with one or more features in other images. Important aspects of matching are metrics and criteria to decide whether two features should be associated, and data structures and algorithms for matching efficiently.

The ideal system will be able to detect a large number of meaningful features in the typical image, and will match them reliably across different views of the same scene / object. Critical issues in detection, description and matching are therefore robustness with respect to viewpoint and lighting changes, the number of features that may be detected in the typical image, the frequency of false alarms and mismatches, and the computational cost of each step. Different applications weigh these requirements differently. For example, viewpoint changes more significantly in object recognition, SLAM and wide-baseline stereo than in image mosaicking, while the frequency of false matches may be more critical in object recognition, where thousands of potentially matching images are considered, rather than in wide-baseline stereo and mosaicing where only few images are present.

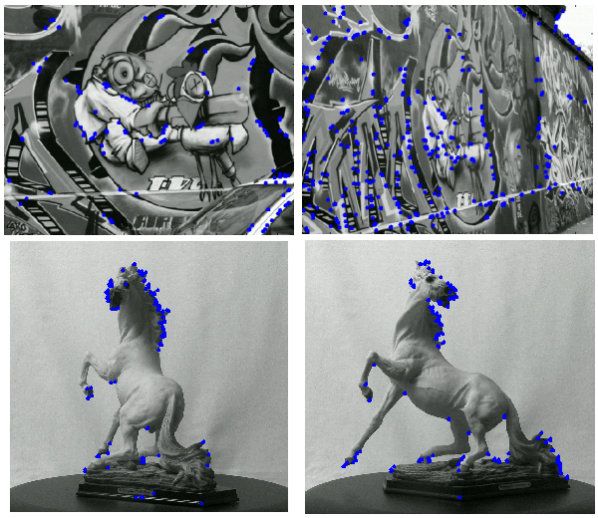

Figure 1: (top row) Important viewpoint change for a flat scene. Many interest points can be matched after the transformation - images from K.Mikolajczyk (bottom row) Similar viewpoint change for a 3D scene. Many points are located in highly textured regions near borders of the object. When the object is rotated, the local geometric structure of the image varies a lot, which makes matching features more challenging because of occlusion and changes in appearance

A number of different feature detectors [2, 7, 8, 9], feature descriptors [10, 11, 12, 13, 14] and feature matchers [5, 6] have been proposed in the literature. They can be variously combined and concatenated to produce different systems. Which combination should be used in a given application? A couple of studies are available. Schmid [5] characterized and compared the performance of several features detectors. Later, Mikolajczik and Schmid [15] focused primarily on the descriptor stage. For a chosen detector, the performance of a number of descriptors was assessed. These evaluations of interest point operators and feature descriptors, have relied on the use of flat images, or in some

cases synthetic images. The reason is that the transformation between pairs of images can be computed easily, which is convenient to establish ground truth. However: many of the applications of feature finders and descriptors is matching across different views of 3D objects and scenes, as in the case of wide-baseline stereo, motion analysis and object recognition.

In the present study we evaluate the performance of feature detector and descriptors for images of 3D objects that are viewed under different viewpoint and lighting contions. To this effect, we collected a database of 100 objects viewed from 144 different calibrated viewpoints under 3 lighting conditions. We also developed a practical and accurate method for establishing automatically ground truth in images of 3D scenes. Another difference with previous studies is that we use a metric for accepting/rejecting feature matches due to D. Lowe [13]; it is based on the ratio of the distance of a given feature from its best match vs the distance to the second best match. This metric has been shown to perform better than the traditional distance-to-best-match.

In section 2 we describe the geometrical considerations which allow us to construct automatically a ground truth for our experiments. In section 3 we describe our laboratory setup, as well as the database of images we collected. Section 4 describes the decision process used in order to assess performances of detectors and descriptors. Section 5 presents and discusses the experiments. Section 6 contains our conclusions.

2 Ground truth

In order to evaluate a particular detector-descriptor combination we need to calculate the probability that a feature which was extracted in a given image can be matched to the corresponding feature in an image of the same object/scene viewed from a different viewpoint. For this to succeed, the physical location must be visible in both images, the feature detector must detect it in both cases with minimal positional variation, and the descriptor of the features must be sufficiently close. In order to compute this probability we must have a ground truth: whenever the matching software proposes a correspondence between two features we must be able to tell whether this correspondence is correct or not. Conversely, whenever a feature is detected in one image, we must be able to tell whether in the corresponding location in another image a feature was detected and whether such feature was matched.

We establish ground truth by using epipolar constraints between triplets of calibrated views of the objects. We distinguish between a 'master' or 'primary' view (A in Fig. 2) a 'test' view B, and an 'auxiliary view' C. Given one feature A_1 in the master image, any feature in B matching the master feature must satisfy the constraint of belonging to the corresponding epipolar line. This excludes most potential matches but not all of them (in our experiments, typically 5-10 features remain out of 300-600). We make the test more stringent by imposing a second constraint. An epipolar line is associated to the master feature in the auxiliary image C. As it will be clear later, the auxiliary viewpoint is close enough to the master viewpoint that a clear correspondence C_1 may be established by matched based on appearance and one epipolar constraint. In turn, C_1 produces an epipolar line in B. The intersection of the primary and auxiliary epipolar lines in B uniquely identifies a small matching region which either contains one feature or none.

The benefit of using the double epipolar constraint in the test image is that any correspondence - or lack thereof - may be validated with extremely low error margins. The cost is that only a fraction of the master features have a correspondence in the auxiliary image, thus limiting the number of features triplets that can be formed. Therefore we are able to measure performance for only a subset of the features detected in the master image. If we call $p_{A_1}(\theta)$ the probability that, given a master feature A_1, a match will exist in a view of the same scene taken from a viewpoint θ degrees apart, the triplet (A_1, B_1, C_1) exists with probability $p_{A_1}(\theta_{AC}) \cdot p_{A_1}(\theta_{AB})$, while the pair (A_1, C_1) exists with higher probability $p_{A_1}(\theta_{AB})$. While the measurements we take allow for a relative assessment of different methods, they need to be renormalized by $1/p_{A_1}(\theta_{AC})$ to obtain absolute performance figures (see section 5).

3 Experimental setup

3.1 Setup of photographic equipment

Our acquisition system consists of 2 cameras taking pictures of objects on a motorized turntable (see Fig. 3). The third camera needed for the 3-pole epipolar constraints discussed above is a virtual camera. For example, if the viewpoint change wished between cameras A and C is 20°, the turntable is rotated by 20°. The pictures acquired by A with this new turntable position, are pretended to be acquired by a third camera that would be 20° degrees apart from A. These 'virtual cameras' save us a considerable amount of time in terms of calibration work: since calibration images can be shared across virtual cameras, we do not need a new set of calibration images for each angle.

Additionally, each acquisition was repeated with 3 lighting conditions in order to evaluate the performance of detectors and descriptors with respect to changes in light. Two photographic spotlights and umbrellas were used on each side of the turntable, the 3 lighting conditions were obtained by switching on one light, then the other, then both of them.

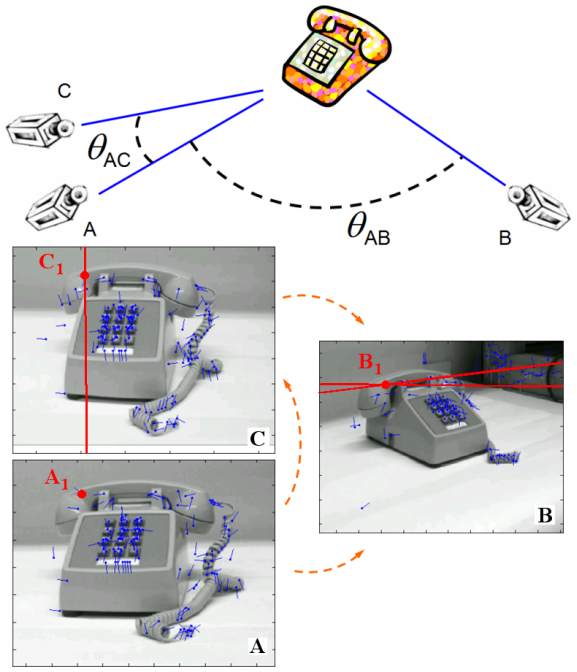

Figure 2: (Top)Diagram of 3-pole epipolar constraints. (Bottom) Example of matching process across 3 views for one feature.

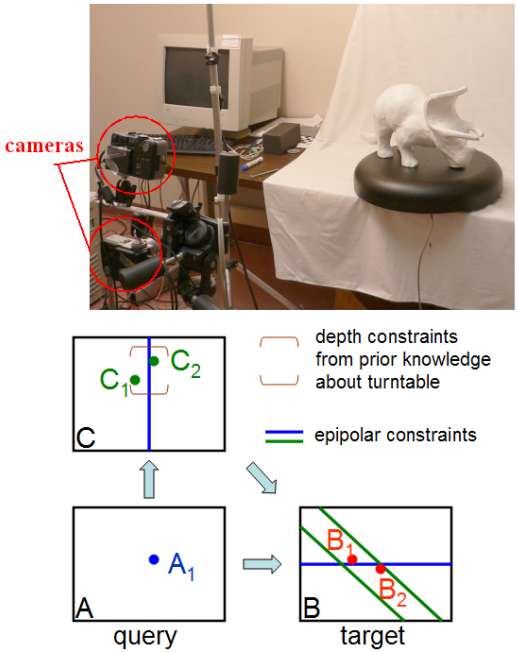

Figure 3: (Top) Photograph of our laboratory setup. Each object was placed on a computer-controlled rotary turntable which could be rotated with $1/50$ degree resolution and 10^{-5} degree accuracy. Two computer-controlled cameras imaged the object. The cameras were located 9° apart with respect to the object. Our cameras have a resolution of 4Mpixels. (Bottom) Diagram explaining the geometry of our three-camera arrangement and of the triple epipolar constraint.

3.2 Calibration

Prior to the objects image acquisition, the cameras need to be calibrated [16]. The calibration images were acquired using a checkerboard pattern placed at different orientations using the turntable. Both cameras were automatically calibrated using a subset of these images and the calibration routines included in Intel's Open CV library. The 'virtual cameras' were calibrated using a different subset of the calibration pictures (an alternative would consist of applying a rotation matrix to the calibration parameters of the 'real' cameras).

Unfortunately, we need to take into account uncertainty on the position of the epipolar line, due to calibration error.

Given two images, the mapping between a point x in the first image and its epipolar line l in the second view can be written $l = Fx$, where F is the fundamental matrix of the stereo system of cameras. Hartley and Zisserman [17] showed that the envelope of the epipolar lines obtained when F varies around its mean value, is a hyperbola, they expressed its parameters in terms of the covariance matrix of the fundamental matrix F.

In practice, we estimated directly the envelope of the epipolar lines with Monte-Carlo simulations using perturbations of the calibration grids. For each run, random calibration images were selected, for each of them a corner of the grid was randomly chosen, and its position was shifted randomly by up to 5 pixels. This quantity was chosen so that it would produce a reprojection error on the grid's corners that was comparable to the one observed during calibration. This was followed by the calibration optimization. For each point of the first image, the Monte-Carlo process leads to a bundle of epipolar lines in the second image, whose envelope is the hyperbola of interest. The envelope should therefore be computed separately for each point of the first image. In our situation, hyperbolas generated by various points from the first image exhibited very similar eccentricities and focal distances. The width between the two branches of the hyperbola varied between 3 and 6 pixels. Since exhaustive computation of all possible envelopes would be too time- and storage-consuming, the same eccentricity and focal distance were used for all points corresponding to a given pair of images.

3.3 Detectors and descriptors

3.3.1 Detectors

- The Harris detector [7] relies on first order derivatives of the image intensities. It it based on the second order moment matrix - also called squared gradient matrix -, which is computed at every location in an image. A cornerness map

is computed based on trace and determinant of this matrix. The points selected as interesting features are the locations of the extrema of this cornerness map.

- The Hessian detector [8] is a second order filter. The corner strength is here the negative determinant of the matrix of second order derivatives (or hessian matrix). The local maxima of the corner strength are taken as interest points. For this detector as well as the Harris detector, multi-scale versions were actually used.

- The Difference-of-gaussian - or DOG - filters [12, 18] the image by a filter consisting of the difference of gaussians with different covariances. The operation is repeated over a range of scales. The selected features are the local maxima in scale and space, of the filtered pyramid of strength maps. This approach fits naturally in the scale-space framework, therefore scale invariance is guaranteed when applying it in a multi-scale approach.

- The Kadir-Brady detector [9] computes maps of the local entropy at different scales. Interest points are extracted at locations that exhibit both a local maxima of entropy over scale, and locations in space where the intensity probability density function varies fastest.

The first and second method are implemented in a multi-scale scheme, so that all methods are scale-invariant.

3.3.2 Descriptors

- Sift features [13] are computed from gradient information. A first step consists of evaluating a main orientation for each feature, in order to obtain orientation-invariance. A second step describes local appearance by histograms of local gradients sorted into 16 locations bins and 8 orientation bins, for a descriptor dimension of 128. The histogramming process ensures a smooth transition of descriptor when location is shifted.

- PCA-Sift [14] computes a primary orientation similarly to Sift. Local patches are then projected onto a lower-dimensional space by using PCA analysis. The PCA projection matrix is learned beforehand from an independent set of patches. At the moment we have only code for its combination with DOG detector (as was described in [14]

- Steerable filters [10] are generated by applying banks of oriented Gaussian derivative filters to an image.

- Differential invariants [5] start from local derivatives of the intensity image (up to 3rd order derivative), and combine them into quantities which are invariant with respect to rotation. Those invariants avoid the problem of computing the major orientation of a patch, which is discarded.

- Spin images [11] represent an overall object as seen from the interest point. The reference orientations are the normal and tangent plane to the surface. The spin descriptor stores histograms of greylevels indexed by the coordinates in the feature's reference frame.

4 Performance evaluation

4.1 Setup and decision scheme

The features detectors and descriptors were evaluated in terms of performance on a keypoint matching problem.

Each feature from a test image was compared with 10^5 features from a database. The database contained both features from one image of the same object (viewed in different lighting conditions or from a different viewpoint), as well as a large number of features extracted from unrelated images.

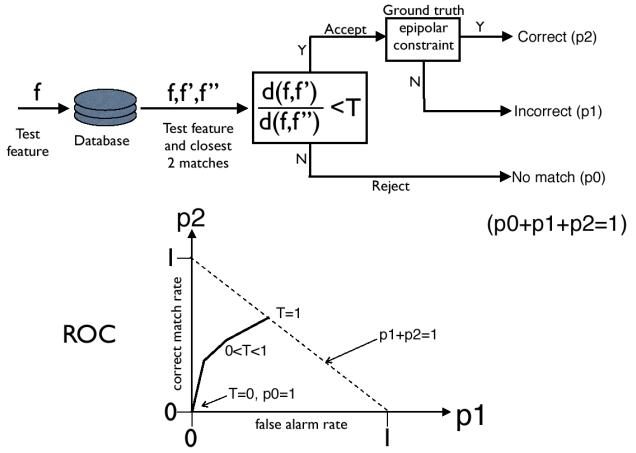

Figure 4: (Top) Diagram showing our decision strategy for the feature matching problem. (Bottom) Result as a ROC

The diagram in Fig. 4 shows the decision strategy. A match to a feature is identified by searching for the closest neighbour to its appearance vector, in a tree containing the whole database (random objects and views of the correct object). Given a threshold T on the quality of the apperance match, the keypoint returned by the search is accepted or rejected. Note that only one potential match is considered for each feature, unlike [15] where multiple database points may be accepted for each query feature.

If the candidate match is accepted, it can be correct, i.e. correspond to the same physical point, or incorrect. If it comes from a wrong image, it is incorrect. If it comes from a view of the correct object, we use the double epipolar constraints with the following method. Starting from A_1 in image A, candidate matches are identified along the corresponding epipolar line in image C. Besides, the object lies on the turntable which has a certain width, so that only a known region on the epipolar line is allowed. There remains n candidate matches $C_1...C_n$ in C (typically 0-4 points). These points generate epipolar lines in B, which intersect the epipolar line from A_1 at points $B_1...B_n$. If the candidate match is one of these points we declare it as a correct match, in the alternative it is considered incorrect.

In case no feature was found along the epipolar line in image C, the initial point A_1 is discarded and doesn't con-

tribute to any statistics, since our inability to establish a triple match is not caused by a poor performance of the detector on the target image B.

Note that this method doesn't guarantee the absence of false alarms. But it offers the important advantage of being purely geometric. Any method involving appearance vectors as an additional constraint would be dependent on the underlying descriptor and bias our evaluation.

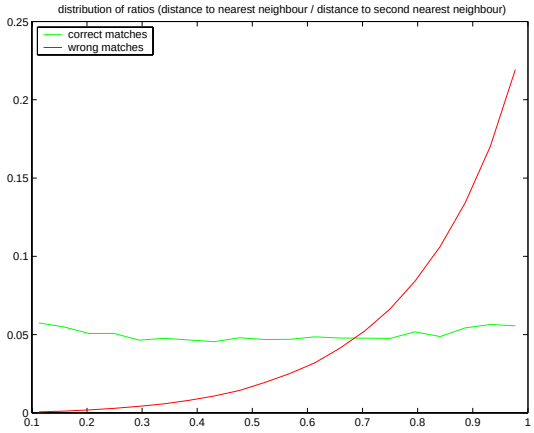

Figure 5: Distributions of the distance ratio between best and second best match for correct correspondences (green) and incorrect matches (red). These curves are similar to the ones in Fig.11 of [13]. Note that the shape of the result for the correct matches is significantly different to Lowe's curve. We do not know yet the reason for this discrepancy. The major difference between his analysis and ours is that we used 3D objects, while he used flat images with added noise.

4.2 Distance measure in appearance space

In order to decide on acceptance or rejection of a candidate match (first decision in Fig. 4), we need a metric on appearance space. Instead of using the euclidean distance as in [15, 14], we use the distance ratio introduced by Lowe [13].

The proposed measure compares the the euclidean distances of the query point to its best and second best matches. In Fig. 4 the query feature and its best and second best matches are denoted by f, f' and f'' respectively. The used criterion is the ratio of these two distances, i.e. $\frac{d(f,f')}{d(f,f'')}$. This ratio characterizes how distinctice a given feature is, and will avoid ambiguous matches. A low value means that the best match performs significantly better than its best contender, and will thus be a reliable match. A high value of the distance ratio will be obtained when the features points are clustered in a tight group in appearance space. Those features are not distinctive enough relatively to each other. In order to avoid a false alarm it is safer to reject the match.

Fig. 5 shows the resulting distribution of distance ratios corresponding to our database. The distance ratios statistics were collected while running our matching problem, these distributions correspond to the (steerable filters / spin) combination. Correct and incorrect matches were identified using the process described in 4.1.

4.3 ROC curve

As seen in the previous section and Fig. 4, the system can have 3 outcomes. In the first case, the match is rejected based on appearance (probability p_0). In the second case, the match is accepted based on appearance, but the geometry constraints are not verified: this is a false alarm (probability p_1). In the third alternative, the match verifies both appearance and geometric conditions, this is a correct detection (probability p_2). These probabilities verify $p_0 + p_1 + p_2 = 1$. The false alarm rate is further normalized by the number of database features (10^5). Detection rate and false alarm rate can be written as

$$falsealarmrate == \frac{\#false\,alarms}{\#attempted\,matches \cdot \#database} \quad (1)$$

while the detection rate is

$$detection\,rate = p_2 = \frac{\#detections}{\#attempted\,matches} \quad (2)$$

5 Results and Discussion

Fig 6 shows the detection results when viewing angle was varied. Each of the first 4 rows displays results when varying the descriptor for a given detector. The last row is a summary displaying for each detector, only the descriptor that performed best.

The left hand side graphs display the ROC curves obtained by varying the threshold T in the first step of the matching process (threshold limiting the distance ratio to first and second best matches). The Sift descriptor performed consistently best with all detectors. The Harris detector obtained the best performance among all feature detectors, although the other combinations (detector / Sift) had a comparable performance. In our graphs the false alarm rate was further normalized by the size of the database (10^5)so that the maximum false alarm rate was 10^{-5}. The right hand side curves show the detection rate as a function of the viewing angle for a fixed false alarm rate of 10^{-6} was chosen (one false alarm every 10 attempts). This false alarm rate corresponds to distinct distance ratio thresholds for each detector / descriptor combination. Those thresholds varied between 0.56 and 0.70 (a bit lower than the 0.8 value chosen by Lowe in [13]). For two detectors (Dog and Kadir/Brady), steerable filters were slightly more reliable,

in the other cases Sift was winning again. Again, the Harris detector was the best detector by a very small margin. Regarding descriptors, with our setting and 3D objects, PCA-Sift didn't seem to outperform Sift as would be expected from [14].

Note that in the stability curves, the detection rate at $0\circ$ is only of the order of $1/3$. This corresponds to the case where the query image and the target image are shots taken exactly at the same position. The large drop in detections is due to fact that the match to the auxiliary image (see section 2) succeeded on average in 1 out of 3 attempts. Since the difference in angle between cameras A and C is known to be 9°, we obtain the value of the angle $p_{A_1}(\theta_{AC})$ mentioned in section 2: $p_{A_1}(\theta_{AC} = 9^\circ) \approx 1/3$. This is consistent with the value of $p_{A_1}(\theta_{AB=10^\circ})$, which is the ratio between the detection rates at 10° and 0° ($\frac{detection(10^\circ)}{detection(0^\circ)} \approx \frac{0.1}{0.3} = 1/3$). If we want to estimate the detection rates that should be obtained for pairwise matching only (without camera C), we can scale the detection curves by a factor of 3.

Another observation concerns the dramatic drop in number of matched keypoints with viewpoint change. For a viewpoint change of 30° the detection rate was of the order of 3%. Even if we rescale by a factor 3, this means that only about 10% of the features can be matched safely.

Fig. 7 shows the results obtained when changing lighting conditions and keeping the viewpoint unchanged. This situation was much easier to handle: since the position of the features shouldn't change, we don't need to introduce the auxiliary image C. The 4 panels on the left display again the ROCs obtained by varying the descriptor for a given detector, while the last panel on the right summarizes the best results and shows only one descriptor for each detector. This time, the steerable filters obtained consistently the best performance, while the best detector seemed to be again the Harris method.

6 Conclusion

This paper presented a new method for evaluating interest point detectors and feature descriptors. Using epipolar constraints, we are able to extract with high reliability ground truth matches from 3D images, instead of using planar surfaces or synthetic images.

The Sift descriptor is the representation that performed best with respect to viewpoint changes, while the representation based on steerable filters was the winner when considering changes in lighting conditions. In both conditions, the multiscale Harris detector performed best in both conditions among the interest point operators that were tested.

Our setup is inexpensive and easy to reproduce in order to collect statistics on correct matches between 3D images. In particular, those statistics will be helpful for probabilistic recognition algorithms, e.g. the ones developed in [20, 21].

7 Acknowledgments

The authors are grateful to David Lowe, Krystian Mikolajczyk, Timor Kadir and Yan Ke for providing part or all of their detectors and descriptors code.

References

[1] T. Tuytelaars and L. Can Gool, "Wide baseline stereo matching based on local affinely invariant regions", in *BMVC*, 2000.

[2] J. Matas, O. Chum, M. Urban, T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions", *BMVC*, 384-393, 2002.

[3] S. Se, D.G. Lowe and J. Little, "Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks", in *IJRR*, 21(8)735-738,2002.

[4] M. Brown and D.G. Lowe, "Recognising panoramas", in *ICCV*, pp. 1218-25, 2003.

[5] C. Schmid and R. Mohr, "Local Greyvalue Invariants for Image Retrieval", *PAMI*, 19(5):530-535, 1997.

[6] D.G. Lowe, "Object Recognition from Local Scale-invariant Features", in *ICCV*, 1150-1157, 1999.

[7] C. Harris and M. Stephens, "A combined corner and edge detector", in *Alvey Vision Conference*, 147-151,1988.

[8] P.R. Beaudet, "Rotationally Invariant Image Operators", in *IJCPR*, Kyoto, Japan, 1978, pp.579-583

[9] T. Kadir, A. Zisserman and M. Brady "An Affine Invariant Salient Region Detector", in *ECCV* 228-241, 2004.

[10] W. Freeman and E. Adelson, "The design and use of steerable filters", in *PAMI*, 13(9):891-906, 1991.

[11] A.E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes", in *PAMI*, pp.433-449, 1999.

[12] J.L. Crowley and A.C. Parker, "A Representation for Shape Based on Peaks and Ridges in the Difference of Low-Pass Transform", *IEEE. Trans. on Patt. Anal. Mach. Int.*, Vol. 6, pp. 156-168, 1984.

[13] D.G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", *IJCV*, 60(2):91-110, 2004.

[14] Y.Ke and R. Sukthankar, "PCA-Sift: A More Distinctive Representation for Local Image Descriptors", in *CVPR*, 2004.

[15] K. Mikolajczyk and C. Schmid "A performance evaluation of local descriptors", *Int. Conf. Comp. Vis. Patt. Recog.*, 2003

[16] J.Y. Bouguet, "Visual methods for three-dimensional modeling", Caltech, 1999.

[17] R. Hartley and A. Zisserman, "Multiple View Geometry in computer vision", Cambridge, 2000.

[18] T. Lindeberg, "Scale-space theory: a Basic Tool for Analising Structures at Different Scales", *J. Appl. Stat.*, 21(2), pp.225-270, 1994.

[19] C. Schmid, R. Mohr and C. Bauckhage, "Evaluation of interest point detectors", *IJCV*, 37(2):151-172,2000

[20] G. Carneiro, A.D. Jepson, "Flexible Spatial Models for Grouping Local Image Features", in *CVPR*, 2004

[21] P. Moreels and P. Perona, "Common-Frame Model for Object Recognition", in *NIPS*, 2004

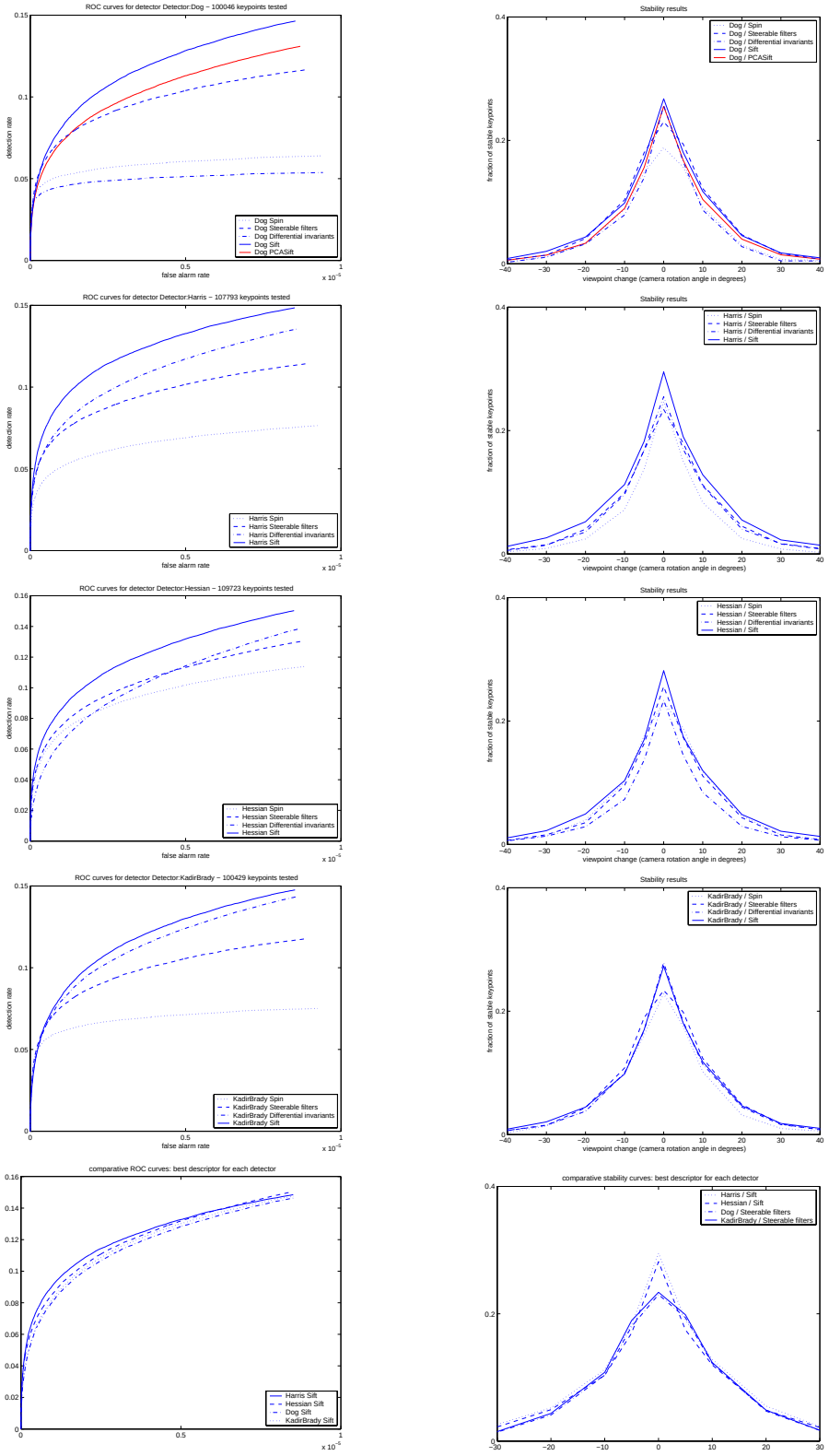

Figure 6: (left)ROC curves showing the performance of combinations of feature finders and coders using constant lighting condtions. The ROC curves were obtained by averaging performance across ten objects and eleven viewpoint differences: $0^0, \pm 5^0, \pm 10^0, \pm 20^0, \pm 40^0, \pm 60^0$. The 0^0 condition was computing using different images taken from the same vantage point after a full rotation of the platform. The bottom plot compares the performance of each coder paired with the detector that performed best with that coder. (right) Corresponding detection rate as a function of change in angle. The false alarm rate was fixed at 1e-6.

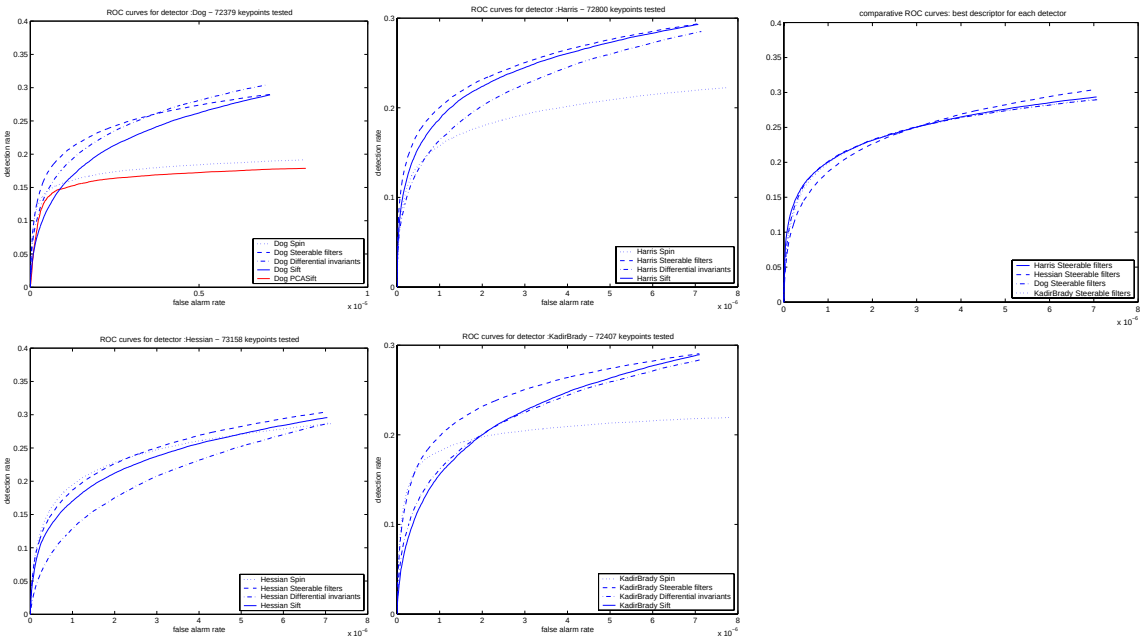

Figure 7: Detection rate as a function of change in lighting conditions. The detection rate is averaged over 10 objects and 3 changes in lighting conditions.

Figure 8: Our calibrated database consists of photographs of 100 objects which were imaged in three lighting conditions: diffuse lighting, light from left and light from right. We chose our objects to represent a wide variety of shapes and surface properties. (Top) Eight sample objects from our collection. (Bottom) Each object was rotated with 5° increments and photographed at each orientation with both cameras and three lighting conditions for a total of $72 \times 2 \times 3 = 432$ photographs per object. Eight such photographs are shown for one of our objects.

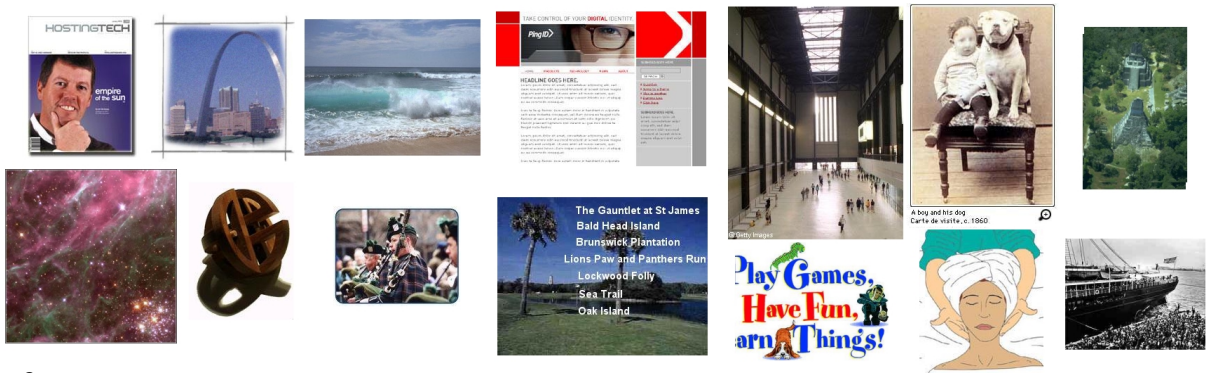

Figure 9: Sample of the images that were used to load the feature database. 535 images were obtained from the Google Image search engine by typing 'things'. Out of the detections generated by these images, 10^5 keypoints were randomly selected and included in our database of features.