Evaluation of Features Detectors and Descriptors based on 3D objects

Pierre Moreels and Pietro Perona
California Institute of Technology, Pasadena CA91125, USA

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 automatically. We collect a database of 100 objects viewed from 144 calibrated viewpoints under three different lighting conditions. We find that the combination of Hessian-affine feature finder and SIFT features is most robust to viewpoint change. Harris-affine combined with SIFT and Hessian-affine combined with shape context descriptors were best respectively for lighting changes and scale 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 for autonomous vehicles [3], mosaicking images [4] and recognizing objects [5, 6]. This operation typically involves three distinct steps. First a 'feature detector' 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.

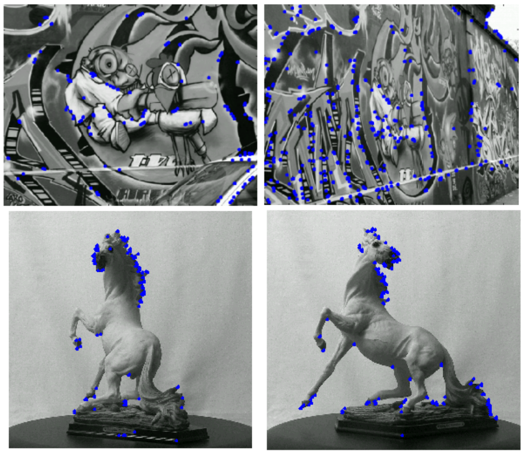

Figure 1: (top row) Large ($\approx 50°$) viewpoint change for a flat scene. Many interest points can be matched after the transformation - images courtesy of K.Mikolajczyk - the appearance change is modeled by an affine transformation. (bottom row) Similar viewpoint change for a 3D scene. Many visually salient features are associated with locations where the 3D surface is irregular or near boundaries, the change in appearance of these features with the viewing direction is not easy to model.

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 robustness with respect to viewpoint and lighting changes, the number of features detected in a 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.

A number of different feature detectors [2, 7, 8, 9, 10, 11], feature descriptors [6, 12, 13, 14] and feature matchers [5, 6, 15, 16] 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. Recently, Mikolajczik and Schmid [17] 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

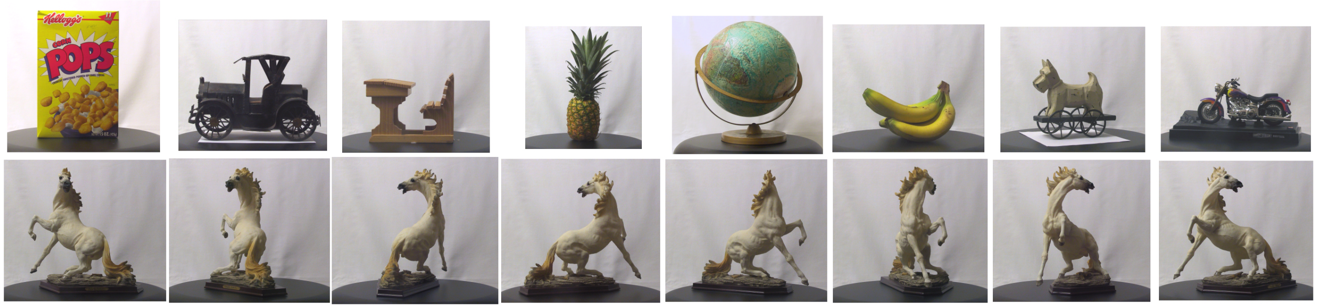

Figure 2: Our calibrated database consists of photographs of 100 objects which were imaged in three lighting conditions: diffuse lighting, light from left and light from right. We chose our objects to represent a wide variety of shapes and surface properties. Each object was photographed by two cameras located above each over, 10° apart.(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.

easily, which is convenient to establish ground truth.

However, the relative performance of various detectors can change when switching from planar scenes to 3D images (see Fig. 12 and [18]). Features detected in an image are generated in part by texture, and in part by the geometric shape of the object. Features due to texture are flat, lie far from object boundaries and exhibit a high stability across viewpoints [5, 17]. Features due to shape are found near edges, corners and folds of the object. Due to self-occlusions, they have a much lower stability with respect to viewpoint change. These features due to shape, or 3D features, represent a large fraction of all detected features.

The present study is complementary to those from [5, 14, 17, 18]. We evaluate the performance of feature detectors and descriptors for images of 3D objects viewed under different viewpoint, lighting and scale conditions. 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. Unlike [18] ground truth is established using geometric constraints only, so that the feature/descriptor evaluation is not biased by an early use of conditions on appearance matches. Besides, our method is fully automated, so that the evaluation can be performed on a large-scale database, rather than on a handful of images as in [17, 18].

Another novel aspect is the use of a metric for accepting/rejecting feature matches due to D. Lowe [6]; 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 and 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 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 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. To compute this probability we must be able to tell if any tentative match between two features 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 matched.

We establish ground truth by using epipolar constraints between triplets of calibrated views of the objects (this is an alternative to using the trifocal tensor [19]). We distinguish between a 'reference' view (A in Fig. 3) a 'test' view C, and an 'auxiliary' view B. Given one feature f^A in the reference image, any feature in C matching the reference feature must satisfy the constraint of belonging to the corresponding 'reference' epipolar line. This excludes most potential matches but not all of them (in our experiments, typically 0-5 features remain out of 300-600). We make the test more stringent by imposing a second constraint. An epipolar line l^B is associated to the reference feature in the auxiliary image B. Again, f^A has typically 5-10 potential matches along l^B, each of which in turn generates an 'auxiliary' epipolar line in C. The intersection of the primary and auxiliary epipolar lines in C identify a small matching regions, in which statistically only zero or one features are detected.

Note that the geometry of our acquisition system (Fig. 3 $\&$ Fig. 4) does not allow the degenerated case where the reference point is on the trifocal plane and both epipolar constraints are superposed.

The benefit of using the double epipolar constraint in the test image is that any correspondence - or lack thereof - may

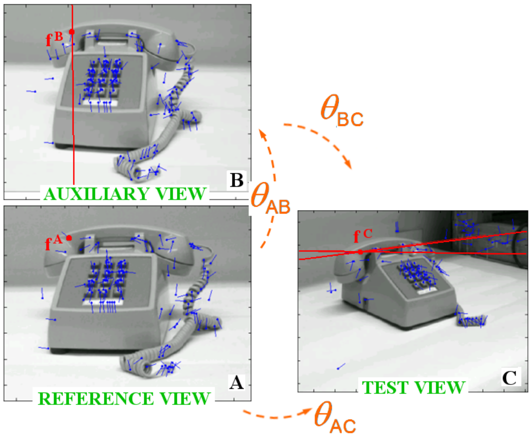

Figure 3: Example of matching process for one feature.

be validated with extremely low error margins. The cost is that only a fraction (50-70%) of the reference features have a correspondence in the auxiliary image, thus limiting the number of features triplets that can be formed. If we call $p_{f^A}(\theta)$ the probability that, given a reference feature f^A, a match will exist in a view of the same scene taken from a viewpoint θ degrees apart, the triplet (f^A, f^B, f^C) exists with probability $p_{f^A}(\theta_{AC}) \cdot p_{f^B}(\theta_{AB})$, while the pair (f^A, f^C) exists with higher probability $p_{f^A}(\theta_{AC})$. While the measurements we take allow for a relative assessment of different methods, they should be renormalized by $1/p_{f^A}(\theta_{AB})$ to obtain absolute performance figures.

3 Experimental setup

3.1 Photographic setup and database

Our acquisition system consists of 2 cameras taking pictures of objects on a motorized turntable (see Fig. 4). The change in viewpoint is performed by the rotation of the turntable. The lower camera takes the reference view, then the turntable is rotated and the same camera takes the test view. Each acquisition was repeated with 3 lighting conditions obtained with a set of photographic spotlights and diffusers.

The database consisted of 100 different objects. Fig. 2 shows some examples from this databaset. Most objects were 3-dimensional, with folds and self-occlusions, which are a major cause of features instability in real-world scenes, as opposed to 2D objects. We included some flat objects (e.g. box of cereals). The database contains both textured objects (pineapple, globe) and objects with a more homogenous surface (bananas, horse).

3.2 Calibration

The calibration images were acquired using a checkerboard pattern. Both cameras were automatically calibrated using the calibration routines in Intel's Open CV library [20].

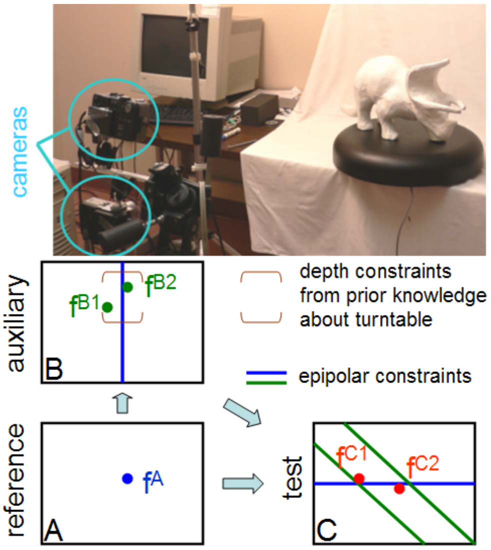

Figure 4: (Top) Photograph of our laboratory setup. Each object was placed on a computer-controlled turntable which can be rotated with $1/50$ degree resolution and 10^{-5} degree accuracy. Two computer-controlled cameras imaged the object. The cameras were located 10° apart with respect to the object.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

Uncertainty on the position of the epipolar lines position was estimated by Monte Carlo perturbations of the calibration patterns. Hartley & Zisserman [21] showed that the envelope of the epipolar lines obtained when the fundamental matrix varies around its mean value, is a hyperbola. The calibration patterns were perturbed 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 P 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 width between the two branches of the hyperbola varied between 3 and 5 pixels. The area inside the hyperbola defines the region allowed for detection of a match to P.

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 (or squared gradient matrix).
- The Hessian detector [8] is a second order filter. The corner strength is here the negative determinant of the matrix of second order derivatives.
- Affine-invariant versions of the previous two detectors

Figure 5: A few examples of the 535 irrelevant images that were used to load the feature database. They were obtained from Google by typing ‘things’. 10^5 features detected in these images were selected at random and included in our database

[10]. The affine rectification process is an iterative warping method that reduces the feature’s second-order moment matrix to have identical eigenvalues.
- The Difference-of-gaussian filters [11] selects scale-space extrema of the image filtered by a difference of gaussians.
- The Kadir-Brady detector [9] selects locations where the local entropy has a maximum over scale and where the intensity probability density function varies fastest.
- MSER features [2] use a watershed flooding process on the image. Regions are selected at locations of slowest expansion of the flooding basins.

3.3.2 Descriptors

- SIFT features [6] are computed from gradient information. Invariance to orientation is obtained by evaluating a main orientation for each feature and offsetting it. Local appearance is then described by histograms of gradients.
- PCA-SIFT [14] computes a primary orientation similarly to SIFT. Local patches are then projected onto a lower-dimensional space by using PCA analysis.
- Steerable filters [12] are generated by applying banks of oriented Gaussian derivative filters to an image.
- Differential invariants [5] combine local derivatives of the intensity image (up to 3rd order derivative) into quantities which are invariant with respect to rotation.
- Shape context descriptors [13] represent the neighborhood of the interest point by color histograms using log-polar coordinates.

4 Performance evaluation

4.1 Setup and decision scheme

The performances of features detectors and descriptors were evaluated on a feature matching problem.

Each feature from a test image was appearance-matched against a large database. The nearest neighbor in this database was selected and tentatively matched to the feature. The database contained both features from one reference image of the same object ($10^2 - 10^3$ features depending on the detector), as well as a large number (10^5) of features extracted from unrelated images. The use of this large database replicates the matching process in applications from object/class recognition.

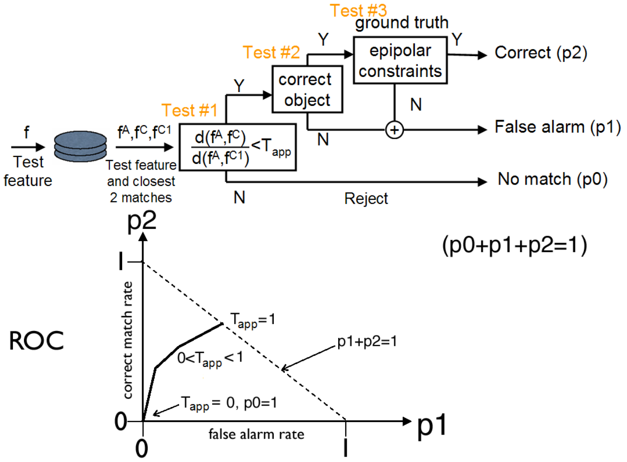

Figure 6: (Top) Diagram showing the process used to classify feature triplets. (Bottom) Conceptual shape of the ROC trading off false alarm rate with detection rate. The threshold T_{app} cannot take values larger than 1 and the ROC is bounded by the curve $p1 + p2 = 1$.

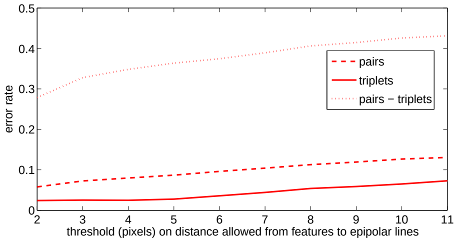

Figure 7: Operator-assisted validation of our automated ground truth. A sample of 3000 pairs and triplets was randomly selected from our dataset. Two experts classified each pair and triplet by hand as to whether it was correct or not. The fraction of wrong triplets is displayed as a function of the maximum distance allowed to the epipolar line (curve ‘triplets’). Our experiments were conducted using adaptive thresholds of 3-5 pixels (see section 3.2), yielding $\approx 2\%$ of incorrect triplets. A method based on a single epipolar line constraint (‘pairs’) would have entailed a rate of wrong correspondences three times higher. In particular, the rate of wrong correspondences is very high for features that could be matched in two images but not in all 3 images (‘pairs − triplets’).

The diagram in Fig.6-top shows the decision strategy. Starting from feature f^A from reference image A, a match to f^A 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). The feature returned by the search is accepted or rejected (Test $\#1$) by comparing the difference in appearance to a threshold T_{app}.

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 (Test #2), it is incorrect. If it comes from a view of the correct object, we use epipolar constraints (Test #3) with the following method (Fig.4-(bottom)). Starting from feature f^A in reference image A, candidate matches are identified along the corresponding epipolar line in the auxiliary image B. Besides, the object lies on the turntable which has a known depth, so that only a known region on the epipolar line is allowed. There remains n candidate matches $f^{B_1} \ldots f^{B_n}$ in B (typically 0-5 points). These points generate epipolar lines in the test image C, which intersect the epipolar line from f^A at points $f^{C_1} \ldots f^{C_n}$. If the candidate match is one of these points we declare it as a correct match, in the alternative it is considered incorrect (false alarm).

In case no feature was found along the epipolar line in the auxiliary image B, the initial point f^A is discarded and doesn't contribute 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 C.

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.

In order to evaluate the fraction of wrong correspondences established by our geometric system, 2 users examined visually random triplets accepted by the system and classified them into correct and incorrect matches. 3000 matches were examined, results are reported in Fig.7(right). The users also classified matches obtained by a simpler method that uses only the reference and test views of the object and one epipolar constraint - cf. section 2 - The fraction of wrong matches is displayed as a function of the threshold on the maximum distance in pixels allowed between features and epipolar lines. We also display the error rate for features that could be matched using the 2-views method, but for which no triplet was identified. The method using 3 views shows a significantly better performance.

4.2 Distance measure in appearance space

In order to decide on acceptance or rejection of a candidate match (first decision in Fig.6), we need a metric on appearance space. Instead of using directly the Euclidean/Mahalanobis distance in appearance as in [17, 14], we use the distance ratio introduced by Lowe [6].

The proposed measure compares the distances in appearance of the query point to its best and second best matches. In Fig.6 the query feature and its best and second best matches are denoted by f^A, f^C and f^{C1} respectively. The criterion used is the ratio of these two distances, i.e. $\frac{d(f^A,f^C)}{d(f^A,f^{C1})}$. This ratio characterizes how distinctice a given feature is, and avoids ambiguous matches. A low value means that the best match performs significantly better than its best contender, and is thus a reliable match. A high value of the distance ratio is 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.

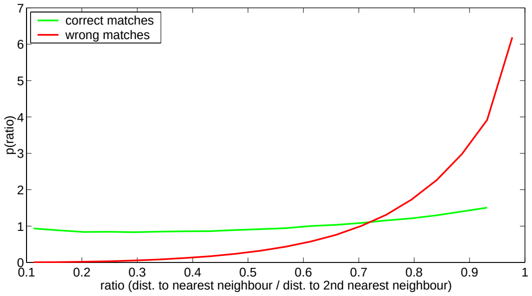

Figure 8: Sample pdf of the distance ratio between best and second best match for correct correspondences (green) and false alarms (red). These curves are analogous to the ones in Fig.11 of Lowe [6]. Lowe's correct-match density is peaked around 0.4 while ours is flat – this may be due to the fact that we use 3D objects, while D.Lowe uses flat images with added noise.

Fig.8 shows the resulting distribution of distance ratios. The distance ratios statistics were collected while running our matching problem. Correct matches and false alarms were identified using the process described in 4.1.

4.3 Detection and false alarm rates

As seen in the previous section and Fig.6, 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

$$false\_alarm\_rate = \frac{\#false\,alarms}{\#attempted\,matches \cdot \#database} \tag{1}$$

$$detection\_rate = \frac{\#detections}{\#attempted\,matches} \tag{2}$$

5 Results and Discussion

Fig.9 shows the detection results when viewing angle was varied and lighting/scale was held constant. Panels a-h display results when varying the feature detector for a given

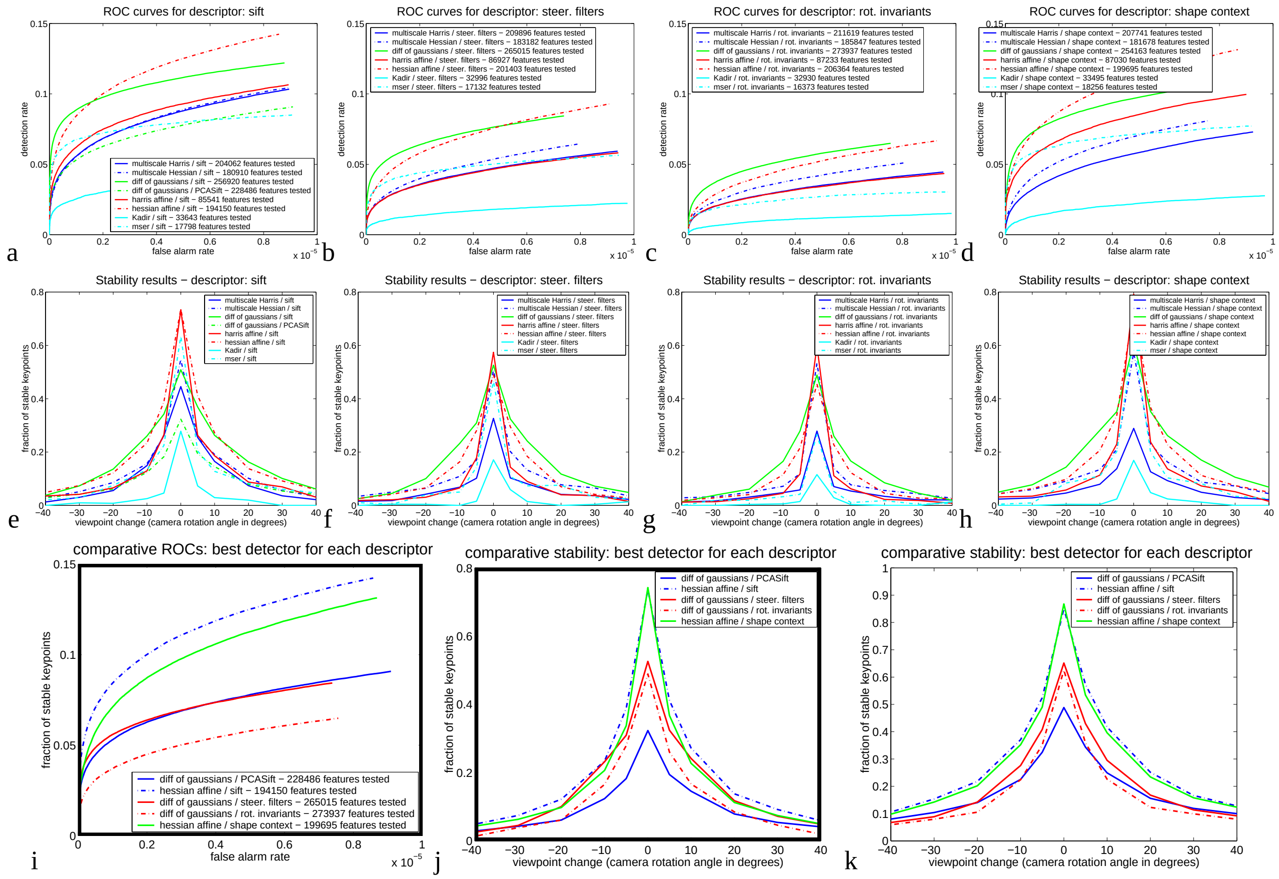

Figure 9: Performance for viewpoint change - each panel a-d shows the ROC curves for a given descriptor when varying the detector. Panels e-h show the corresponding stability rates as a function of the rotation angle. The 0° result is obtained with different images from the same location. Panels i-j show the combination of each descriptor with the detector that performed best for that descriptor. Panel k is similar to panel j, but the database used for the search tree contained only the features extracted from the correct image (easier task).

image descriptor. Panels i-j summarize for each descriptor, the detector that performed best. Panels a-d display the ROC curves obtained by varying the threshold T_{app} in the first step of the matching process (threshold on distinctiveness of the features' appearance). The number of features tested is displayed in the legend. Panels e-h 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 different 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 [6]).

The Hessian-affine and difference-of-gaussians detectors peformed consistently best with all descriptors. While the absolute performance of the various detectors varies when they are coupled with different descriptors, their rankings vary very little. The combination of Hessian-affine with SIFT and shape context obtained the best overall score, with the advantage to SIFT. In our graphs the false alarm rate was normalized by the size of the database (10^5) so that the maximum false alarm rate was 10^{-5}. The PCA-SIFT descriptor is only combined with difference-of-gaussians, as was done in [14]. PCA-SIFT didn't seem to outperform SIFT as would be expected from [14].

In the stability curves, the fraction of stable features doesn't reach 1 when $\theta = 0°$. This is due to several factors: first, triplets can be identified only when the match to the auxiliary image succeeds (see section 2). The $10°$ viewpoint change between reference and auxiliary image prevents a number of features to be identified in both images.

Another reason lies in the tree search. The use of a tree that contains both the correct image and a large number of unrelated images replicates the matching process used in recognition applications. However, since some features have low distinctiveness, the correct image doesn't collect all the matches. In order to evaluate the detection drop due to the search tree, the experiment was run again with a search tree that contained only the features from the correct image. Fig.9-k shows the stability results, the performance is 10-15% higher.

A third reason is the noise present in the camera. On repeated images taken from the same viewpoint, this noise

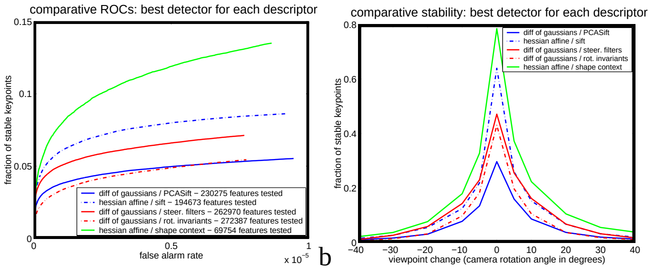

Figure 10: Results for viewpoint change, using the Mahalanobis distance instead of the Euclidean distance on appearance vectors

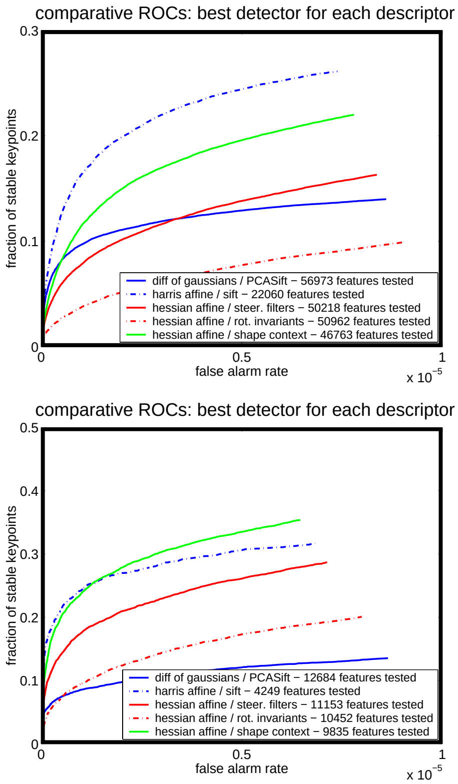

Figure 11: (Top) ROCs for variations in lighting conditions. Results are averaged over 3 lighting conditions. (Bottom) ROCs for variations in scale.

causes $5 - 10\%$ of the features to be unstable.

Another observation concerns the dramatic drop in number of matched features with viewpoint change. For a viewpoint change of $30°$ the detection rate was below 5%.

Fig.10 shows the results ('summary' panel only) when the Euclidean distance on appearance descriptors is replaced by the Mahalanobis distance. Most relative performances were not modified. Hessian-affine performed again best, while shape context and SIFT were the best descriptors. In this case, shape context outperformed SIFT.

Fig.11(top) shows the results obtained when changing lighting conditions and keeping the viewpoint unchanged. This task is easier: since the position of the features shouldn't change, we don't need to introduce the auxiliary image B. Only the 'summary' panel with the best detector for each descriptor are displayed. This time, the combination which achieved best performance was Harris-affine combined with SIFT.

Fig.11(bottom) displays the results for a change of scale. The scale change was performed by switching the camera's focal length from 14.6mm to 7.0. Again, the figure displays only the 'summary' panel. Hessian-affine combined with shape context and Harris-affine combined with SIFT obtained the best results.

6 Conclusion

We compared the most popular feature detectors and descriptors on a benchmark designed to assess their performance in recognition of 3D objects. In a nutshell: we find that the best overall choice is using an affine-rectified detector [10] followed by a SIFT [6] or shape-context descriptor [13]. These detectors and descriptor were the best when tested for robustness to change in viewpoint, change in lighting and change in scale. Amongst detectors, runner-ups are the Hessian-affine detector [10], which performed well for viewpoint change and scale change, and the Harris-affine detector [10], which performed well for lighting change and scale change.

Our benchmark differs from previous work from Mikolajczyk & Schmid in that we use a large and heterogeneous collection of 100 3D objects, rather than a handful of flat scenes. We also use Lowe's ratio criterion, rather than absolute distance, in order to establish correspondence in appearance space. This is a more realistic approximation of object recognition. A major difference with their findings is a significantly lower stability of 3D features. Only a small fraction of all features (less than 3%) can be matched for viewpoint changes beyond $30°$. Our results favoring SIFT and shape context descriptors agree with [17]. However, regarding detectors, all affine-invariant methods don't seem to be equivalent as suggested in [23], e.g. MSER performs poorly on 3D objects while it is very stable on flat surfaces.

We find significant differences in performance with respect to a previous study on 3D scenes [18]. One possible reason for these differences is the particular statistics of their scenes, which appear to contain a high proportion of highly textured quasi-flat surfaces (boxes, desktops, building facades, see Fig.6 in [18]). This hypothesis is supported by the fact that our measurements on piecewise flat objects (Fig.12) are more consistent with their findings. Another difference with their study is that we establish ground truth correspondence purely geometrically, while they use appearance matching as well, which may bias the evaluation.

An additional contribution of this paper is a new method for establishing geometrical features matches in different views of 3D objects. Using epipolar constraints, we are able to extract with high reliability (2% wrong matches) ground truth matches from 3D images. This allowed us to step up detector-descriptor evaluations from 2D scenes to 3D objects. Comparing to other 3D benchmarks, the ability to rely on an automatic method, rather than painfully acquired manual ground truth, allowed us to work with a large number of heterogeneous 3D objects. Our setup is inexpensive and easy to reproduce for collecting statistics on correct matches between 3D images. In particular, those statistics will be helpful for tuning recognition algorithms such as [6, 15, 16]. Our database of 100 objects viewed from 72 positions with three lighting conditions and the full 3D ground truth will be available on our web site.

7 Acknowledgments

The authors are grateful to Timor Kadir, Yan Ke, David Lowe, Jiri Matas and Krystian Mikolajczyk 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”, *ICCV*, 2003.

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

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

[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] K. Mikolajczyk and C. Schmid, “An affine invariant interest point detector”, *ECCV*, 2002.

[11] 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.

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

[13] S. Belongie, J. Malik, J. Puzicha, “Shape Matching and Object Recognition Using Shape Contexts”, in *IEEE PAMI*, 2002.

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

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

[16] P. Moreels and P. Perona, “Common-Frame Model for Object Recognition”, in *NIPS*, 2004

[17] K. Mikolajczyk, C. Schmid. “A performance evaluation of local descriptors”, To appear, *IEEE Trans. on Patt. Anal. and Mach. Int.*

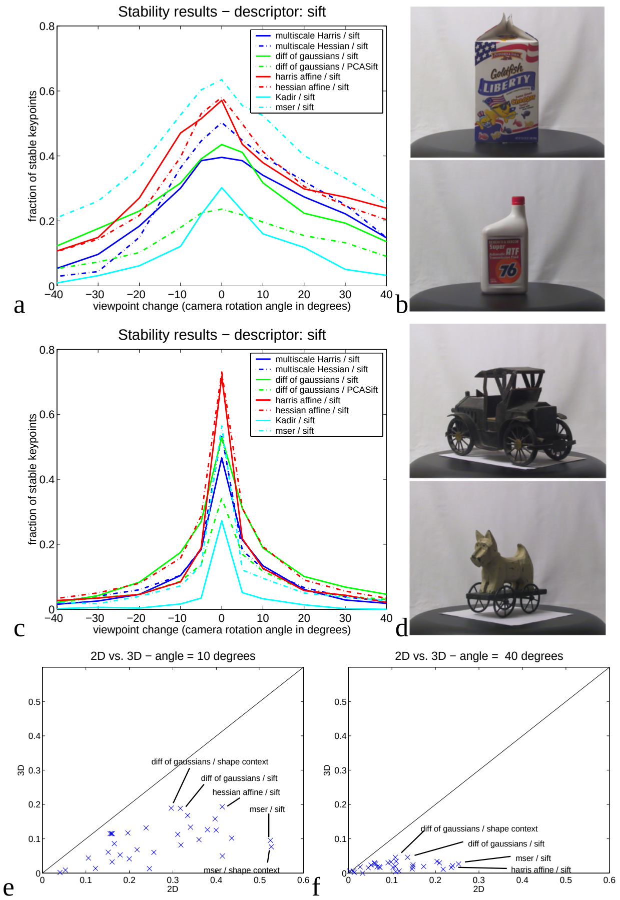

Figure 12: Flat vs. 3D objects - Panel a. shows the stability curves obtained for SIFT for the piecewise flat objects in panel b. Similarly, panel c. shows the SIFT stability curves for the 3D objects in panel d. ‘Flat’ features are significantly more robust to viewpoint change. Panels e-f show the fractions of stable features for the same piecewise 2D objects versus the same 3D objects, for all combinations of detectors / descriptors in this study. Scatter plots are displayed for rotations of 10° and 40°. A few combinations whose relative performance changes significantly are highlighted.

[18] F. Fraundorfer and H. Bischof “Evaluation of local detectors on non-planar scenes”, *OAGM/AAPR*, 2004

[19] A. Shashua and M. Werman, “On the trilinear tensor of three perspective views and its underlying geomtry”, *ICCV*, 1995.

[20] J.Y. Bouguet, “Visual methods for three-dimensional modeling”, PhD thesis, Caltech, 1999.

[21] R. Hartley and A. Zisserman, “Multiple View Geometry in computer vision”, Cambridge editor, 2000.

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

[23] K. Mikolajczyk et al., “A Comparison of Affine Region Detectors”, *submitted, IJCV*, 2004.