Local Shape Estimation from a Single Keypoint

Alberto Del Bimbo, Fernando Franco and Federico Pernici
Media Integration and Communication Center (MICC), University of Florence, Italy
{delbimbo,franco,pernici}@dsi.unifi.it

Abstract

This paper presents a novel approach to estimate local homography of points belong to a given surface. While others works attempt this by using iterative algorithms developed for template matching, our method introduces a direct estimation of the transformation. It performs the following steps. First, a training set of features captures appearance and geometry information about keypoints taken from multiple views of the surface. Then incoming keypoints are matched against the training set in order to retrieve a cluster of features representing their identity. Finally the retrieved clusters are used to estimate the local pose of the regions around keypoints. Thanks to the high accuracy, outliers and bad estimates are filtered out by multiscale Summed Square Difference (SSD) test.

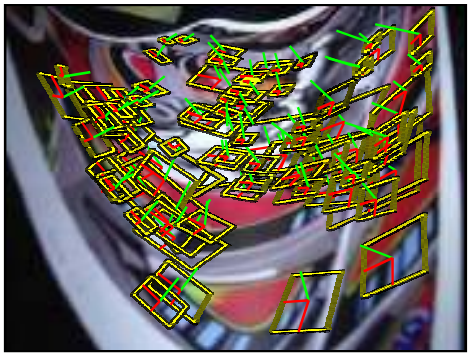
1. Introduction

The last years have seen the development of many affine region detectors that derive an approximation of the local image transformation around points of interest. Matching is performed using a region descriptor which provides invariance by getting rid of most of the complexity due to the image transformation. More recently, an approach based on the learning instead of specific detectors has been proposed [5, 4, 6]. This method appears to be faster and more reliable, but relies on iterative refinements that makes it unqualified for very large image database. In this paper, we propose a new approach that performs the other way around by direct estimation of the local homography around points of interest. Given a reference image of the target surface and an input image containing this surface, our method proceeds in three steps. We first generate a training set of features that compactly captures geometry and appearance information about multiple views of the same keypoints. Then input keypoints are associated with the correspondent sets of features by a matching process and a geometry consistency checking. Finally, the informations related to keypoints in the sets are used to estimate the local perspective transformations. Outliers and bad estimates are filtered out using a multiscale SSD validation. As shown in Fig. 1, our approach avoids specific estimation of the transformation and gives us a more reliable estimate than affine region detectors for both planar and non-rigid surfaces. The rest of the paper is organized as follows. Sect. 2 outlines the state of the art about local pose estimation. Sect. 3 contains an overview of the approach with details about the training set generation and the local homography estimation. In Sect. 4 experimental results are shown and discussed. Conclusions are drawn in Sect. 5.

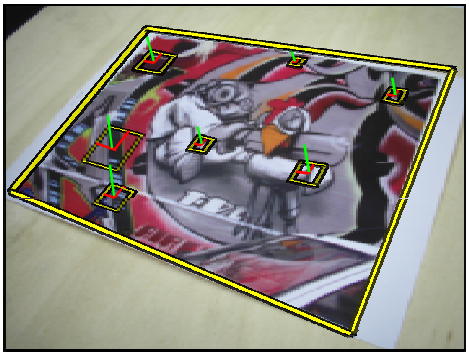
2. Related Works

Many computer vision applications rely on the recovery of properties of interest points, or keypoints. For example, retrieving the poses of keypoints in addition to matching them is a fundamental task in vision-based robot localization [3], object recognition [14] or image retrieval [13] to transform unconstrained problems into geometrically constrained ones. The standard approach proceeds by first using some particular affine region detectors and by then using SIFT descriptors computed on the rectified regions to match the points. Many different detectors have been proposed in the recent years. Among them, the Hessian-Affine detector of Mikolajczyk and Schmid [10] and the MSER detector by Matas et al. [9] have been shown to be the most reliable ones. However, they retrieve only an affine transformation without estimating the full perspective pose and often require handcrafting the descriptors to achieve insensitive to specific kind of distortion. Recently, a novel class of learning-based methods that attemps to compute local homography of a planar patch around keypoint has been developed [5, 4, 6, 12]. In particular the approaches of [5, 4, 6] mainly consist in two steps: the incoming point of interest is matched against a database of keypoints, each of which is associated to a coarse estimation of its pose (defined as the homography between a reference patch and the patch centered on the point); the coarse pose retrieved is hence iteratively refined by applying the template matching techniques in [7] and the result is successively refined with a

(a)



(b)



(c)

Figure 1. Approach overview. (a): Given a training image, we use SIFT keypoints to perform local patch pose estimation. The results are very accurate and mostly free of outliers in the case of non-rigid surfaces (b) and planar objects (c). To better appreciate accuracy, surface normals are plotted according to internal camera parameters

iterative template matching algorithm [2]. For the first step, [5] uses the Ferns classifier [11] while [4, 6] relies on linear classifiers.

In [12] keypoints are detected at 2D corners and matched to a pre-defined set of corners. Differently from the previous approaches, an estimation by regression is inserted here in the loop of the Ferns classifier for matching. In this way, the homographic transformation is directly checked during matching. Finer regression is only performed for close matches.

3. The Approach

Given a point of interest extracted at run-time, we want to match it against a training set of features and to accurately estimate its local pose represented by a homography. Our approach performs in three steps. The first builds a training set of features which captures geometry and appearance information about keypoints taken from multiple views of a given 3D object. The second step matches an incoming point of interest against the database in order to retrieve a cluster of features representing keypoint identity. In the last step the retrieved cluster is used to estimate the local patch pose. Thanks to the high accuracy, outliers and bad estimates are then filtered out by multiscale Summed Square Difference (SSD) test.

3.1. Training Set Generation

Let us consider a set of m 3D points of interest $\mathbf{K} = \{\mathbf{k}_i\}_{i=1}^{m}$ lying on the surface of a given object. The aim is to build a large training set of features which captures geometry and appearance about different patches around these points extracted by multiple views of the object. According to this, an effective method to build the training set is to generate random synthetic views of the object using simple geometrical technique and extract SIFT keypoints from them. In this way, we can easily associate each keypoint with information about patch around it and select keypoints that are more stable under noise and perspective distortion. We discuss below the construction of multiple views of the object given a reference image and then the process of extracting and selecting keypoints.

3.1.1 Multiple views sampling.

Under the assumption of local smooth surface patches surrounding points of interest $\mathbf{k}_i$ can be considered as locally planar and their distortion under prospective projection can be represented by homographies. Therefore only one reference image $\mathbf{I}_r$ of the target object could be enough to generate the set of multiple views $\{\mathbf{I}_j\}$. Considering that for moderate foreshortening keypoints keep stable even under some viewpoint changes distorted image views are created

from the reference image $\mathbf{I}_r$ taking a rectangular window of approximately one half the image area around each corner of the reference image, selecting one point at random in each window, and assuming these points as the vertices of the newly generated image $\mathbf{I}_j$ where the original content is warped. Instead, since for strong foreshortening keypoints keep stable only for small variations of the viewing angle, in order to provide a finer sampling, the same procedure is applied to the vertices of already distorted images with windows of approximately one tenth of the image area. Fig. 2 shows some views generated by this process. Generated images are clipped to a given image resolution in order to avoid processing very large images that can emerge in foreshortening condition.

3.1.2 Features extraction.

Once the multiple views $\{\mathbf{I}_j\}$ are sampled, we can extract SIFT keypoints from them in order to associate each feature with geometry and appearance information: $\mathbf{f}_i = \{\mathbf{d}_i, \mathtt{H}_{rj}\}$. Geometry information is captured by the homography $\mathtt{H}_{rj}$ between the reference image and the view $\mathbf{I}_j$ from which the keypoint is taken, while appearance information is represented by SIFT descriptor $\mathbf{d}_i$.

3.1.3 Features selection.

Because of noise and perspective distortion, the points lying on the object surface don't have the same probability $p(\mathbf{k}_i)$ to be found in a target image $\mathbf{I}_t$ in which they are visible at runtime. In order to select the $\mathbf{m}$ keypoints with highest probability to be extracted, we proceed as follows. Let $\mathtt{H}_{rj}$ the homography which transforms the reference image $\mathbf{I}_r$ in the image $\mathbf{I}_j$ which contains the keypoint $\mathbf{k}_{ij}$. By applying $\mathtt{H}_{rj}^{-1}$ to $\mathbf{k}_i$ the found 2D point is back-projected in the coordinate system of $\mathbf{I}_r$ and feed a point accumulator which allows to estimate the probability $p(\mathbf{k}_i)$ with which the corresponding 3D points can be detected in a new image. The 3D points accumulating most votes are retained as points of interest, having a large probability to be detected by SIFT in unknown target images.

3.2. Matching and Homography estimation

Given a set $\mathcal{K}_Q$ of SIFT extracted by an image at runtime, we want to retrieve the identities of keypoints lying on the surface of the target object and to obtain an estimation of their local homography. The problem of retrieving identity can be defined as a search for a function $\mathbf{B} : \mathcal{K}_Q \rightarrow \mathcal{K}_I \cup \mathbf{k}_0$ that assigns to every $\mathbf{k}_q \in \mathcal{K}_Q$ either a cluster[1] of features $\mathbf{C}_q \subset \mathcal{K}_I$ or $\mathbf{k}_0$ representing no matching. According to this, since the training set contains multiple views of each

[1]The term cluster here refer to a group slightly different descriptors obtained in correspondence of a keypoint of the target object.

(a)

(b)

Figure 2. Training set generation. All views are synthesized using a frontal image of the Graffiti as reference image. **(a)** Views synthesized using the first image of Graffiti sequence as reference image. **(b)** Views synthesized using an expanded set of reference image in order to take into account foreshortening perspective effect.

3D point of interest, each keypoint $\mathbf{k}_q \in \mathcal{K}_Q$ is matched to its k nearest neighbors. This can be done in logarithmic time by using a kd–tree to find the approximate nearest neighbors [1]. We use $k = \frac{|\mathcal{K}_I|}{v}$, where v is the number views in the training set. Wrong matched clusters are discarded by checking the likelihood ratio. In particular the cluster $\mathbf{C}_q$ is associated to a keypoint $\mathbf{k}_q$ only if the descriptor of the second-closest neighbor is far enough ϵ to the descriptor of the closest neighbor [1]:

$$\frac{\min_{\mathbf{d}_i \in \mathbf{C}_q} \|\mathbf{d}_q - \mathbf{d}_i\|_2}{\min_{\mathbf{d}_i \in \mathbf{C}_q \setminus \mathbf{B}(\mathbf{d}_q)} \|\mathbf{d}_q - \mathbf{d}_i\|_2} < \epsilon \qquad (1)$$

where

$$\mathbf{B}(\mathbf{d}_q) = \arg \min_{\mathbf{d}_i \in \mathbf{C}_q} \|\mathbf{d}_q - \mathbf{d}_i\|_2 \quad (2)$$

is the Euclidean nearest neighbour of $\mathbf{d}_q$ (in our experiments we used a distance ratio greater than 0.75 as rejection criterion). Since each one of the retrieved keypoints in $\mathbf{C}_q$ has its homography associated, matching also permits to obtain a number of coarse estimation of the keypoint local pose. However checking the geometry consistency of keypoints of cluster $\mathbf{C}_q$ is necessary to filter out wrong matches.

3.2.1 Geometry checking.

Because of visual repeated structure, each cluster $\mathbf{C}_q$ is processed in order to reject false matching with the corresponding $\mathbf{k}_q$. Since the closest neighbor $\mathbf{f}_1$ (i.e. 1-NN) corresponds with high probability to a different view of $\mathbf{k}_q$, we proceed as described in the following pseudo-code:

1. Back-project the location of $\mathbf{f}_1$ in the coordinate system of $\mathbf{I}_r$ using $\mathtt{H}_1^{-1}$.
2. Define a circle c of radius 3 pixels centered on the back-projection of $\mathbf{f}_1$.
3. For each feature $\mathbf{f}_i$ $i = 2..k$ apply steps 4 and 5.
4. Back-project $\mathbf{f}_i$ in the coordinate system of $\mathbf{I}_r$ using $\mathtt{H}_i^{-1}$.
5. Discard $\mathbf{f}_i$ if its back-projection is outside the circle c.

In practice each element of the cluster is validated by checking with its associated homography. Fig. 3 shows an example of that process in a difficult case of visual repeated structure in small localized region (i.e. the concentric pattern of eye) where appearance and the imaged location do not discriminate the surface element identity.

The approach differs from [14] where under the assumption of rigid objects the geometry checking is applied to all the the pairs of keypoints (i.e. the existence of epipolar constraint and/or planarity relationship between 3D surface patches). Since only local information of the keypoint is used with no assumption on object rigidity, the method can be also applied to the case of non-rigid objects and computational requirements are drastically reduced.

3.2.2 Local pose estimation.

After discarding outliers, the local homography of the patch around $\mathbf{k}_q$ is directly estimated using informations associated to the n remaining features. Let $\{\mathbf{d}_i\}_{i=1}^n$ the set of descriptors representing appearance information and $\{\mathbf{h}_i\}_{i=1}^n$ the set of homographies capturing geometry information about these features. The estimate is performed by simply averaging the homographies. For better accuracy, the

(a)

(b)

Figure 3. Geometry checking process. **(a)** The back-projected features outside the circle are discarded as outliers. **(b)** Multiple keypoints taken from the same cluster.

contribution of each homography is weighed according to the NN-distance between the relative descriptor and the descriptor $\mathbf{d}_q$:

$$\hat{\mathtt{H}}(\mathbf{d}) = \frac{1}{n} \sum_{\mathbf{h}_i \in \mathbf{C}_q} \mathbf{w}_i \mathbf{h}_i \quad (3)$$

where $\mathbf{w}_i = \|\mathbf{d}_q - \mathbf{d}_i\|_2$. Fig. 1 shows an application of this estimation process.

3.2.3 Multiscale SSD-based validation

A final validation is needed to remove bad estimated keypoints. Thanks to the accuracy of the retrieved transformations, we are able to reject keypoints using the Summed

Square Difference between the estimated patch and the warped patch in the reference image. We adopt a method similar to the one described in [8] in order to decide at which scale the reference patch should be warped to. In particular we apply warping using a matrix $\mathtt{A}$ computed from the Jacobian of the estimated homography evaluated at the keypoint coordinate (x_0, y_0):

$$\mathtt{A} = \begin{pmatrix} \frac{\partial \hat{\mathtt{H}}}{\partial x} & \frac{\partial \hat{\mathtt{H}}}{\partial y} \\ \frac{\partial \hat{\mathtt{H}}}{\partial x} & \frac{\partial \hat{\mathtt{H}}}{\partial y} \end{pmatrix}_{(x_0, y_0)} . \quad (4)$$

The determinant of matrix $\mathtt{A}$ corresponds to the area (in square pixel), that a single source pixel would occupy in the full-resolution image of the reference view $\mathbf{I}_r$. While $det(\mathtt{A})/4$ is the corresponding area in pyramid level one, and so on. The target pyramid level l is chosen so that $det(\mathtt{A})/4^l$ is closest to unity, basically we attempt to match the warped patch in the pyramid level which most closely matches its scale in the reference view using normalized SSD.

4. Experimental Results

Several experiments were performed in order to assess the effectiveness of the method and compare it against specific affine region detectors. To this end, we generate synthetic views $\mathbf{I}_j$ with a factor of foreshortening ranging from 0 to 70. In this context, factor of foreshortening is a function of the homography $\mathtt{H}_j$ which transforms the vertices of the reference image in the new vertices: $K_{rj} = (\lambda_1^j \lambda_2^j - 1)$, where $\lambda_{1,2}^j$ are the two singular values of $\mathtt{H}_{rj}$. In particular K_j is much greater then 0 as $\mathbf{I}_j$ is a more slanted version than the reference image under perspective transformation. For each view $\mathbf{I}_j$, we apply our method to identify approximately 50 keypoints and retrieve their pose. We repeat this test 2000 times for each cost and report the accuracy results in figure 4, in which our method is denoted by 'SIFTHomography'. To create these graphs we proceed as follow. In the case of affine region detectors, we fit a square tangent to the normalized regions and warp this square back with the inverse transformation to get a quadrangle. In the case of our method, the quadrangle is simply taken to be the patch borders after warping the square on the reference image by the retrieved homography. In Fig. 4(a) we compare the average overlap between the quadrangles obtained using the ground truth homography and those obtained with our method and with affine region detectors. This overlap is very close to 90% for our method, about 5% better than MSER and about 15% than other affine region detectors. Fig. 4(b) shows the comparison of the mean reprojection error for the quadrangle corners. The error of the patch corner is less than four pixel in average and outperforms other methods.

Our current implementation runs at about 15 frame per second using 400.000 features in the database (organized as a vocabulary tree) extracting about 200 SIFT keypoints in the input images, on a standard notebook with an Intel Centrino Core Duo with 2.4GHz and 3Gb RAM.

5. Conclusion and Future Works

This paper introduced a novel method for estimating the local homography of a given 3D object. The effectiveness of our approach relies on two key ideas. First, the generation of a training set that captures geometry and appearance information about multiple views of the same keypoints, and second, the usage of the average operator for the estimate. We have shown that this process avoids specific estimation of the local transformation and gives better results than standard affine region detectors. Since we used only SIFT keypoints, our future work will investigate the use of different detectors and descriptors.

References

[1] J. S. Beis and D. G. Lowe. Indexing without invariants in 3d object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(10):1000–1015, 1999.

[2] S. Benhimane and E. Malis. Homography-based 2d visual tracking and servoing. *Int. J. Rob. Res.*, 26(7):661–676, 2007.

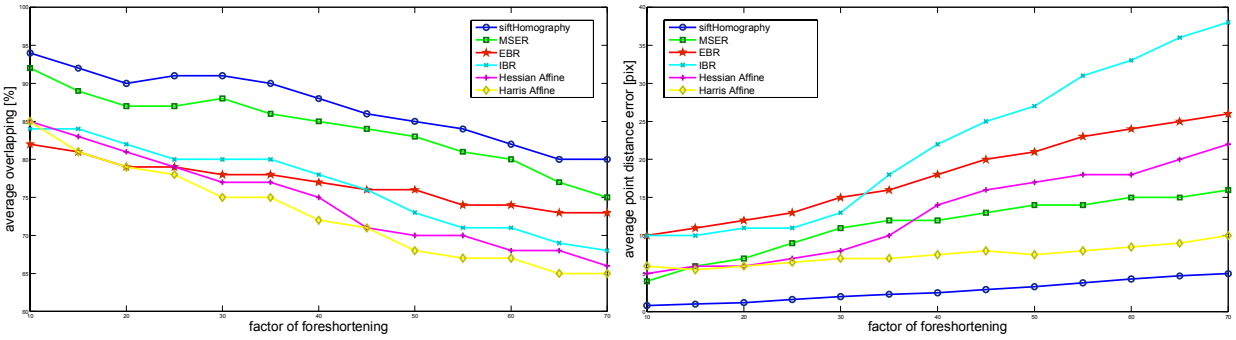
[3] V. Ferrari, T. Tuytelaars, and L. Gool. Simultaneous object recognition and segmentation from single or multiple model views. *Int. J. Comput. Vision*, 67(2):159–188, 2006.

[4] S. Hinterstoisser, S. Benhimane, , V. Lepetit, and N. Navab. Simultaneous recognition and homography extraction of local patches with a simple linear classifier. In *BMVC British Machine Vision Conference 2008*, 2008.

[5] S. Hinterstoisser, S. Benhimane, N. Navab, P. Fua, and V. Lepetit. Online learning of patch perspective rectification for efficient object detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008.

[6] S. Hinterstoisser, O. Kutter, N. Navab, P. Fua, and V. Lepetit. Real-time learning of accurate patch rectification. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2009.

[7] F. Jurie and M. Dhome. Hyperplane approximation for template matching. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(7):996–1000, 2002.



(a) (b)

Figure 4. Comparing our method against affine region detectors. **(a)** Average overlapping area of all correctly matched regions. **(b)** Average sum of the distances from the ground truth for the corner points.

[8] G. Klein and D. Murray. Parallel tracking and mapping for small AR workspaces. In *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07)*, Nara, Japan, November 2007.

[9] J. Matas, O. Chum, U. Martin, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *Proceedings of British Machine Vision Conference (BMVC02)*, 2002.

[10] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1/2):43–72, 2005.

[11] M. Özuysal, P. Fua, and V. Lepetit. Fast keypoint recognition in ten lines of code. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007.

[12] A. Pagani and D. Stricker. Learning local patch orientation with a cascade of sparse regressors. In *Proc. British Machine Vision Conference (BMVC 2009)*, London, UK, September 2009.

[13] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[14] F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce. 3d object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. *Int. J. Comput. Vision*, 66(3):231–259, 2006.