7

Stereo Correspondence with Local Descriptors for Object Recognition

Gee-Sern Jison Hsu
National Taiwan University of Science and Technology
Taiwan

1. Introduction

Stereo correspondence refers to the matches between two images with different viewpoints looking at the same object or scene. It is one of the most active research topics in computer vision as it plays a central role in 3D object recognition, object categorization, view synthesis, scene reconstruction, and many other applications. The image pair with different viewpoints is known as stereo images when the baseline and camera parameters are given. Given stereo images, the approaches for finding stereo correspondences are generally split into two categories: one based on sparse local features found matched between the images, and the other based on dense pixel-to-pixel matched regions found between the images. The former is proven effective for 3D object recognition and categorization, while the latter is better for view synthesis and scene reconstruction. This chapter focuses on the former because of the increasing interests in 3D object recognition in recent years, also because the feature-based methods have recently made a substantial progress by several state-of-the-art local (feature) descriptors.

The study of object recognition using stereo vision often requires a training set which offers stereo images for developing the model for each object considered, and a test set which offers images with variations in viewpoint, scale, illumination, and occlusion conditions for evaluating the model. Many methods on local descriptors consider each image from stereo or multiple views a single instance without exploring much of the relationship between these instances, ending up with models of multiple independent instances. Using such a model for object recognition is like matching between a training image and a test image. It is, however, especially interested in this chapter that models are developed *integrating* the information across multiple training images. The central concern is how to extract local features from stereo or multiple images so that the information from different views can be integrated in the modeling phase, and applied in the recognition phase. This chapter is composed of the following contents:

1. Affine invariant region detection in Section 2. Many invariant image features are proposed in the last decade. Because these features are invariant to image variations in viewpoint, scale, illumination, and other variables, they serve well for establishing stereo correspondences across images. Those with better invariance to viewpoint changes are of special interest as they can be of direct use in the development of object models from stereo or multi-view.

2. Local region descriptors in Section 3: These descriptors transform affine invariant regions into vectors or distributions so that some distance measure can be applied to discern the similarity or difference between features. Again, those with better invariance to viewpoint changes are especially interested.
3. Object modeling and recognition using local region descriptors from multi-view in Section 4: A couple methods are reviewed that develop models by combining the information from local descriptors extracted across multiple views. These methods offer good examples on how to integrate local invariant features across different views.
4. A case study on performance evaluation and benchmark databases in Section 5: Implementation of others' methods for performance comparison with one's own proposed method takes a tremendous amount of time and efforts. Therefore, a database commonly accepted for performance benchmark is needed, and different methods can be evaluated on the same testbed. A performance evaluation example is reviewed with an introduction on its database, followed by a snapshot on other databases also good for study on 3D object recognition using stereo correspondences.

2. Affine regions for stereo correspondence

Affine-invariant region detectors can identify the affine-invariant regions on multiple images which are the projections of the same 3D surface patches. The regions are also considered as *covariant* with geometric and photometric transformations, as the regions detected in one image can be mapped onto those detected in the other using these transformations. Different affine detectors give different local regions in terms of different locations, sizes, orientations and the numbers of detected regions.

Mikolajczyk et al. (2005) have evaluated six affine region detectors, including Harris-affine, Hessian-affine, edge-based region, intensity extrema-based region, salient region and maximally stable extremal region (MSER). This evaluation focuses on the performance of matching between two images with variations caused by viewpoint, scale, illumination, blur and JPEG compression. The detectors for regions only covariant to similarity transform are excluded in their evaluation, for example the interest regions extracted to develop the Scale-Invariant Feature Transform (SIFT) by Lowe (1999; 2004) and the scale invariant features by Mikolajczyk & Schmid (2001). However, the SIFT descriptor (Lowe, 1999; 2004) is used in this evaluation to characterize the intensity patterns of the regions detected by the above six detectors.

The scope of this chapter is on finding stereo correspondences for object recognition, subject to the requirement that the object's model is built on at least a pair of stereo images with different viewpoints. In certain cases, the objects in stereo or multiple images may appear slightly different in scale. Therefore the detectors that perform better than others in rendering correct matches under viewpoint and scale changes are of special interest in this chapter. This performance can be justified by the *repeatability* and *matching score* from the evaluation in Mikolajczyk et al. (2005). It is shown that the Harris-affine detector, Hessian-affine detector and the maximally stable extremal region (MSER) detector are three promising ones in offering reliable stereo correspondences under viewpoint and scale changes. Note that illumination changes, blur and JPEG compression are among the major challenging parameters when recognizing a test image, the three aforementioned detectors also perform well when testing against these parameters, as revealed by Mikolajczyk et al. (2005).

2.1 Harris and hessian affine detectors

Harris affine region detector exploits a combination of Harris corner detector, Gaussian scale-space and affine shape adaptation. The core part is based on the following second moment matrix,

$$M(\mathbf{x},\sigma_D,\sigma_I) = \sigma_D^2 G(\sigma_I) * \begin{bmatrix} L_x^2(\mathbf{x},\sigma_D) & L_x L_y(\mathbf{x},\sigma_D) \\ L_x L_y(\mathbf{x},\sigma_D) & L_y^2(\mathbf{x},\sigma_D) \end{bmatrix} \quad (1)$$

where $L(;\sigma_D)$ is the image smoothed by a Gaussian kernel with differentiation scale σ_D; $L_x(\mathbf{x},\sigma_D)$ and $L_y(\mathbf{x},\sigma_D)$ are the first derivatives of the image along $x-$ and $y-$ directions, respectively, at point $\mathbf{x}$. The derivatives are then averaged in a neighborhood of $\mathbf{x}$ by convolving with $G(\sigma_I)$, a Gaussian filter with integration scale σ_I. The eigenvalues of $M(\mathbf{x},\sigma_D,\sigma_I)$ measure the changes of the gradients along two orthogonal directions in that neighborhood region. When the change is larger than a threshold, the region is considered a corner-like feature in the image.

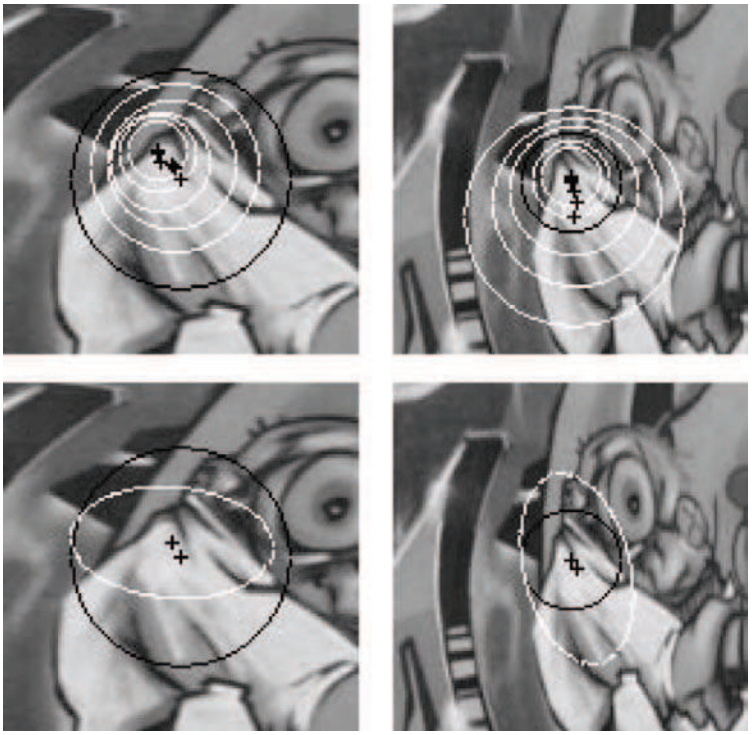


Fig. 1. Scale invariant interest point detection in affine transformed images: (Top) Initial interest points detected by multi-scale Harris detector with characteristic scales selected by Laplacian scale peak (in black–Harris-Laplace). (Bottom) Characteristic point detected with Harris-Laplace (in black) and the corresponding point from the other image projected with the affine transformation (in white). Reproduced from Mikolajczyk & Schmid (2004).

Given an image, the algorithm for detecting Harris affine regions consists of the following steps (Mikolajczyk & Schmid, 2002; 2004; Mikolajczyk et al., 2005):

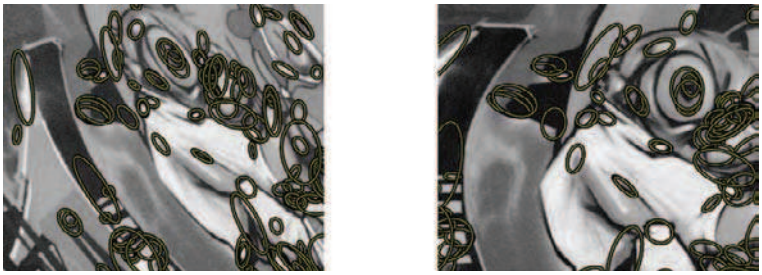
1. *Detection of scale-invariant interest regions using the Harris-Laplace detector and a characteristic scale selection scheme*: Given σ_I and σ_D, the scale-adapted Harris corner detector using the second moment matrix M in (1) can be used to estimate corner-like features. To determine the characteristic scale, σ_I^*, the scale-adapted Harris corner is first applied with a number of preselected scales, resulting in corners in multiple scales. Given these corners, the algorithm given by Lindeberg (1998) can be applied, which iteratively searches for both the characteristic scale σ_I^* and the spatial location $\mathbf{x}^*$ that maximize the Laplacian-of-Gaussians (LoG) over the preselected scales.
2. *Normalization of the scale-invariant interest regions obtained in Step 1 using Affine Shape Adaptation*: The obtained scale-invariant interest regions are normalized using affine shape adaptation (Lindeberg & Gårding, 1997), which again uses the second moment matrix M in (1) but generalized with non-uniform Gaussian kernels for anisotropic regions (versus the uniform Gaussian kernels in (1) for isotropic regions). It is an extension of the regular scale-space obtained by convolution with *rotationally symmetric* Gaussian kernels to an affine Gaussian scale-space obtained by *shape-adapted* Gaussian kernels. This step results in initial estimates on the affine regions.
3. *Iterative estimation of the affine region*: The step in each iterative loop are composed of the generation of a reference frame using a shape adaptation matrix $U^{(k-1)}$, the selection of an appropriate integration scale $\sigma_I^{(k)}$ and differentiation scale $\sigma_D^{(k)}$, and the spatial localization of an interest point $\mathbf{x}^{(k)}$, where $\cdot^{(k)}$ denotes for the k-th iteration. The shape adaptation matrix is the concatenation of square roots of the second moment matrices and is often initialized by the identity matrix. The integration scale is selected at the maximum over a predefined range of scales of the normalized Laplacian, and the differentiation scale is selected at the maximum of normalized isotropy. To reduce complexity, Mikolajczyk & Schmid (2002; 2004) make $\sigma_D = s\sigma_I$, where s is a constant factor between 0.5 to 0.75.
4. Affine region update using the updated scales, $\sigma_I^{(k)}$ and $\sigma_I^{(k)}$, and spatial localizations $\mathbf{x}^{(k)}$. This allows the second moment matrix $M^{(k)}$ renewed, and the shape adaptation matrix $U^{(k)}$ updated.
5. Return to Step 3 if the stopping criterion on the isotropy measure is not met. Because the above algorithm in each iterative loop searches for the shape adaptation matrix $U^{(k)}$ that transforms an anisotropic region into an isotropic region, the iteration terminates when the ratio between the minimum and maximum eigenvalues of $M^{(k)}$ becomes sufficiently close to 1.

Fig. 1, reproduced from Mikolajczyk & Schmid (2004), shows an example from initial estimates of the regions using multi-scale Harris detector to the final affine invariant regions. In addition to the above Harris-Affine region detector based on the Harris-Laplace detector in (1), a similar alternative is Hessian-Affine region detector based on the Hessian matrix (Mikolajczyk et al., 2005),

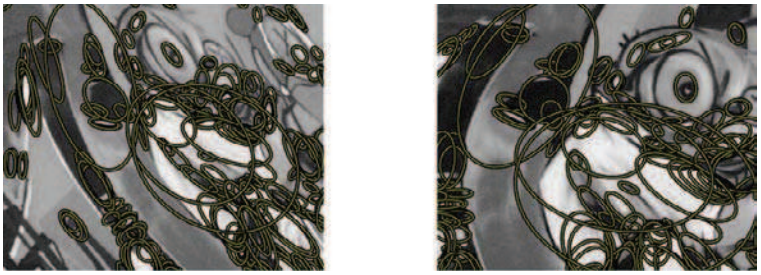
$$H(\mathbf{x}, \sigma_D) = \begin{bmatrix} L_{xx}(\mathbf{x}, \sigma_D) & L_{xy}(\mathbf{x}, \sigma_D) \\ L_{xy}(\mathbf{x}, \sigma_D) & L_{xx}(\mathbf{x}, \sigma_D) \end{bmatrix} \tag{2}$$

According to Mikolajczyk et al. (2005), the second derivatives, L_{xx}, L_{xy} and L_{xy} give strong responses on blobs and ridges. The scheme is similar to the blob detection given by Lindeberg (1998). The points maximizing the determinant of the Hessian matrix will penalize long

structures with small second derivatives in one particular orientation. A local maximum of the determinant indicates the presence of a blob. The detection of Hessian-Affine regions is almost the same as the iterative algorithm for Harris-Affine regions, but with the second moment matrix in (1) replaced by the Hessian matrix in (2). Fig. 2, given in Mikolajczyk et al. (2005), shows examples of Harris-Affine and Hessian-Affine regions.



(a) Harris-Affine



(b) Hessian-Affine

Fig. 2. Examples of regions detected by Harris-Affine and Hessian-Affine detectors; reproduced from Mikolajczyk et al. (2005)

2.2 Maximally stable extremal region (MSER)

MSER is proposed by Matas et al. (2002) to find correspondences between two images of different viewpoints. The extraction of MSER considers the set of all possible thresholds able to binarize an intensity image $I(\mathbf{x})$ into a binary image $E_{t_M}(\mathbf{x})$,

$$E_{t_M}(\mathbf{x}) = \begin{cases} 1 & if I(\mathbf{x}) \leq t_M \\ 0 & otherwise. \end{cases} \tag{3}$$

where t_M is the threshold. A MSER is a connected region in $E_{t_M}(\mathbf{x})$ with little change in its size for a range of thresholds. The number of thresholds that maintain the connected region similar in size is known as the *margin* of the region. One can successively increase the threshold t_M in (3) to detect dark regions, denoted as MSER+; or invert the intensity image first and then increase the threshold to detect bright regions, denoted as MSER-. An example given by Forssén & Lowe (2007) with margin larger than 7 is shown in Fig. 3.

Because it is defined exclusively by the intensity function in the region and the outer border, and the local binarization is stable over a large range of thresholds, the MSER possesses the following characteristics which make it favorable in many cases (Matas et al., 2002; Nistér & Stewénius, 2008):

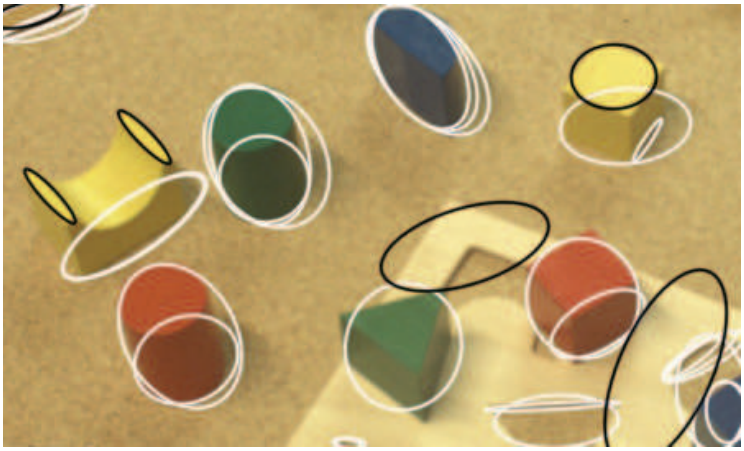


Fig. 3. Regions detected by a MSER with margin 7, reproduced from Forssén & Lowe (2007).

- The regions are closed under continuous (and thus projective) transformation of image coordinates, indicating that they are affine invariant regardless if the image is warped or skewed.
- The regions are closed under monotonic transformation of image intensities, reflecting that photometric changes have no effect on these regions, so they are robust to illumination variations.
- The regions are stable because their support is virtually unchanged over a range of thresholds.
- The detection performs across multiple scales without any smoothing involved, so both fine and large structures are discovered. If it operates with a scale pyramid, the repeatability and the number of correspondences across scales can be further improved.
- The set of all extremal regions can be enumerated in worst-case $O(n)$, where n is the number of pixels in the image.

Besides, the extensive performance evaluation by Mikolajczyk et al. (2005) shows the following characteristics of MSER:

- Viewpoint change: MSER outperforms other detectors in both the original images and those with repeated texture motifs.
- Scale change: MSER is outperformed by the Hessian-Affine detector only, in the repeatability percentage and matching score when the scale factor is large than 2.
- Illumination change: MSER gives the highest repeatability percentage.
- Region size - MSER appears to render more small regions than many others do, and small interest regions can be better in recognizing objects with occlusion.
- Blur - The performance of MSER degrades substantially when blur increases, and therefore, other detectors should be considered when recognizing objects in blur images. This might be the only variable that MSER cannot handle well.

MSER has been extended to color images by Forssén & Lowe 2007. This extension studies successive time-steps of an agglomerative clustering of color pixels. The selection of time-steps is stabilized against intensity scalings and image blur by modeling the distribution of edge magnitudes. The algorithm contains an edge significance measure based on a Poisson image noise model, yielding a better performance than the original MSER from Matas et al. (2002), especially when extracting such interest regions from color images.

3. Local region descriptors

Local region descriptors are mostly in vector forms that can characterize the pattern of an interest point with its neighboring region. Ten different descriptors are reviewed and evaluated by Mikolajczyk & Schmid (2005), including the scale invariant feature transform (SIFT) by Lowe (2004), gradient location and orientation histogram (GLOH) by Mikolajczyk & Schmid (2005), shape context (Belongie et al., 2002), PCA-SIFT (Ke & Sukthankar, 2004), spin images (Lazebnik et al., 2003), steerable filters (Freeman & Adelson, 1991), differential invariants (Koenderink & van Doom, 1987), complex filters (Schaffalitzky & Zisserman, 2002), moment invariants (Gool et al., 1996) , and cross-correlation of sampled pixel values (Mikolajczyk & Schmid, 2005). Five region detectors are used to offer interest regions in this evaluation study: Harris corners, Harris-Laplace regions, Hessian-Laplace regions, Harris-Affine regions and Hessian-Affine regions. Given an image, these detectors are first applied to identify interest regions, which are used to compute the descriptors.
Similar to the previous section that selects the affine invariant regions good for handling viewpoint and scale variations, this section focuses on the region descriptors good for the same variables. Fig. 4, reproduced from Mikolajczyk & Schmid (2005), shows a few comparisons on viewpoint and scale changes in terms of $1-precision$ versus *recall*. $1-$precision and recall are defined as follows:

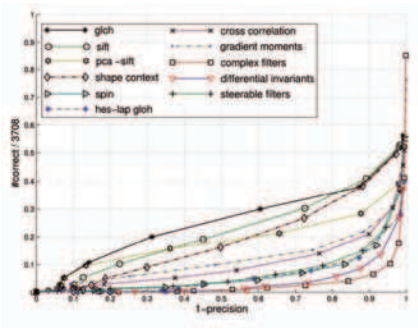
$$1 - precision = \frac{N_f}{N_c + N_f} \tag{4}$$

$$recall = \frac{N_c}{N_{cr}} \tag{5}$$

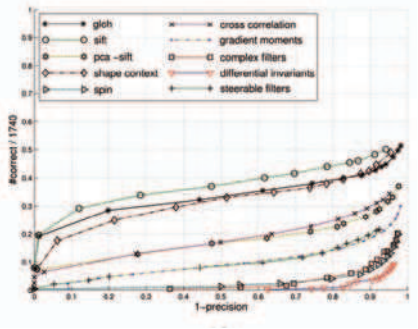
where N_c and N_f are the numbers of correct and false matches, respectively, and both change with the threshold that measures the distance between descriptors. N_{cr} is the number of correspondences. N_c and N_{cr} depend on the overlap error, which measures how well the corresponding regions fit each other under homography transformation. A perfect descriptor would give a unity recall for any precision. In practice, recall increases with decreasing precision (and thus increasing $1-$precision). For any fixed precision, the descriptors that yield higher recalls are more desirable.
It can be seen that GLOH (Mikolajczyk & Schmid, 2005) performs the best, closely followed by SIFT (Lowe, 2004) and shape context (Belongie et al. 2002) in generating more correct matches under viewpoint and scale changes. Actually, as revealed by the extensive experimental study in Mikolajczyk & Schmid (2005), these three descriptors also outperform the others in most tests with other variables.

3.1 SIFT and GLOH descriptors

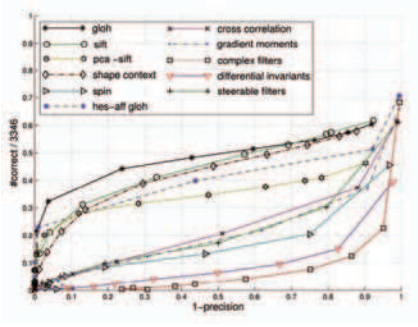
SIFT (Scale-Invariant Feature Transform) descriptor, proposed by Lowe (2004), is derived from a 3D histogram of gradient location and orientation. GLOH (Gradient Location and



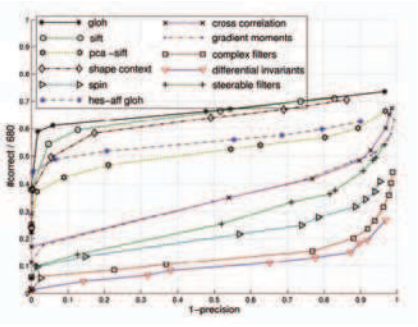
(a) Viewpoint change with structured scene and Hessian-Affine regions



(b) Viewpoint change with textured scene and Hessian-Affine regions



(c) Scale change with structured scene and Hessian-Laplace regions



(d) Scale change with textured scene and Hessian-Laplace regions

Fig. 4. Performance comparison of region descriptors for viewpoint and scale changes, reproduced from Mikolajczyk & Schmid (2005).

Orientation Histogram) is a modified version of SIFT, given by Mikolajczyk & Schmid (2005), which computes a SIFT descriptor for a log-polar location grid with bins in both radial and angular directions.

Figs. 5a and 5b summarizes the computation of a SIFT descriptor. The gradient magnitudes and orientations are first computed at each sample point in a region around an interest point (or *keypoint* as called in Lowe, 2004), as the arrows shown in Fig. 5a. Each arrow shows the magnitude of the gradient by its length, and the orientation by its arrowhead. A Gaussian blur window, shown by the blue circle in Fig. 5a, is imposed on the interest region with σ equal to one half the width of the region's scale, assigning a weight to the magnitude of each sample point. This Gaussian window can avoid sudden changes in the descriptor with small perturbations on the position of the region, and weaken the contribution from the gradients far from the center of the region. Fig. 5a shows a 2×2 descriptor array with 4 subregions inside, and each subregion is formed by 4×4 elements. The gradients in each subregion can be segmented according to the eight major orientations, and summed up in magnitude for each orientation, transforming the 8×8 gradient patterns to the 2×2 descriptor patterns, as shown in Fig. 5b. This 2×2 descriptor pattern gives a vector of $2 \times 2 \times 8 = 32$ in dimension. However,

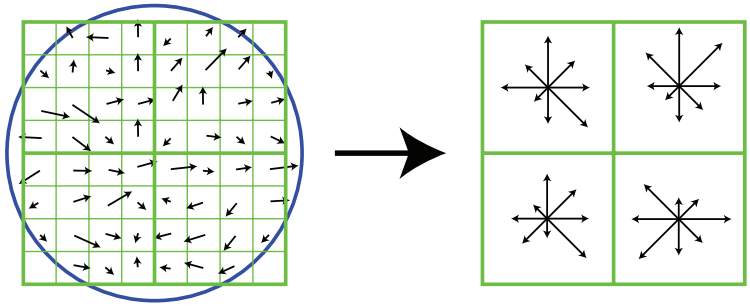


Fig. 5. (a) 2×2 descriptor array with 4 subregions inside, and each subregion is formed by 4×4 elements. The gradients are smoothed by a Gaussian window shown in blue circle. (b) The 8 orientation bins in each subregion can be combined with bins from other subregions, leading to a vector descriptor for this interest region.

based on the experiments by Lowe (2004), the best descriptor that has been exhaustively tested is with 4×4 array, leading to a descriptor vector of $4 \times 4 \times 8 = 128$ in dimension. To obtain illumination invariance, this descriptor is normalized by the square root of the sum of squared components.

GLOH is SIFT descriptor computed for a log-polar location grid with three spatial elements in radial direction (with radius 6, 11, and 15) and eight orientations. Only the subregion with smallest radius is not segmented to orientations, and this gives $2 \times 8 + 1 = 17$ subregion in total. The gradient orientations in each subregion are quantized into 16 bins, and this gives to the interest region a vector of 272 in dimension. PCA (Principal Component Analysis) is then applied to downsize its dimension to 128 using the principal components extracted from 47,000 patches collected from various images. The experiments in Mikolajczyk & Schmid (2005) reveal that GLOH performs slightly better than SIFT in many tests.

3.2 Shape context descriptor

Shape context, proposed by Belongie et al. (2002), is a descriptor that characterize the shape of an object. Given a shape, which can be obtained by an edge detector, one can pick a point p_i out of the n points on the shape and compute the histogram h_i of the relative coordinates of the remaining $n - 1$ points,

$$h_i(k) = \#\{q \neq p_i : (q - p_i) \in bin(k)\} \tag{6}$$

where k denotes for the k-th bin of the histogram, q denotes a point on the shape. This histogram, measured in a *log*-polar space, defines the shape context descriptor of p_i. It reveals the distribution of the shape relative to p_i in terms of $\log(r)$ and θ, where r measures the distance and θ measures the orientation. This design makes the descriptor more sensitive to the locations of nearby shape points than to those farther apart. Belongie et al. (2002), use 5 bins for $\log(r)$ and 12 bins for θ, giving a descriptor of dimension 60; while in Mikolajczyk & Schmid (2005), r is split into 9 bins with θ in 4 bins, resulting in a descriptor of dimension 36. Fig.6, from Belongie et al. (2002), shows an example of shape context computation and matching.

Given a point p_i on the first shape and a point q_i on the second shape, C_{ij}, which denotes the cost of matching these two points, can be computed using their shape context descriptors as

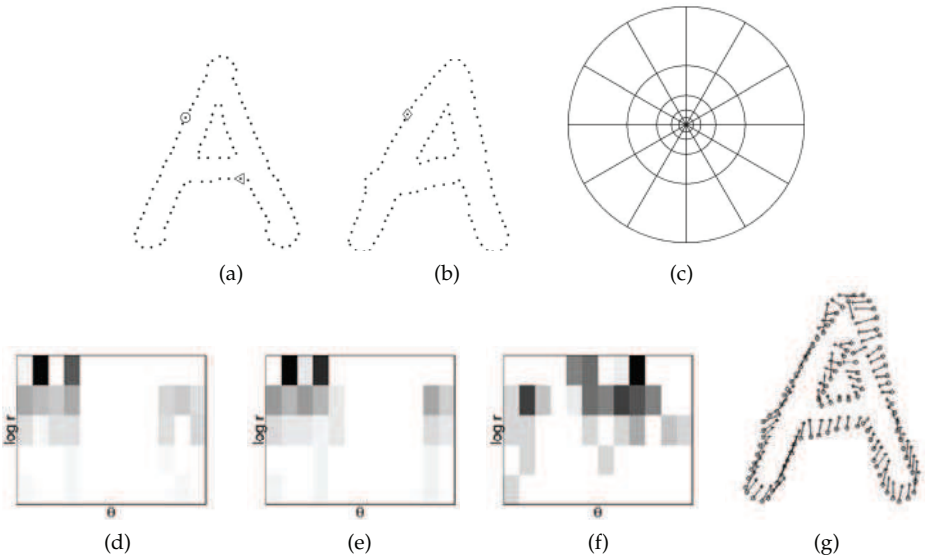


Fig. 6. Shape context computation and matching, (a) and (b) are the sampled edge points of two "A" shapes. (c) Diagram of log-polar histogram bins used for computing shape contexts, 5 bins for $\log r$ and 12 for θ. (d), (e) and (f) are the shape contexts obtained for the reference points marked by $\circ$, $\diamond$, and $\triangleleft$, respectively. Similar patterns between $\circ$ and $\diamond$, and a different one at $\triangleleft$ can be observed. (g) Correspondences found by bipartite matching. All are reproduced from Belongie et al. 2002.

follows,

$$C_{ij} \equiv C(p_i, q_j) = \frac{1}{2} \sum_{k=1}^{K} \frac{|h_i(k) - h_j(k)|^2}{h_i(k) + h_j(k)} \tag{7}$$

where $h_i(k)$ and $h_j(k)$ denote the K-bin normalized histogram at p_i and q_j, respectively. (7) applies the χ^2 test for measuring the difference between distributions. The total cost of matching all point pairs can then be written as

$$H(\pi) = \sum_i C(p_i, q_{\pi(i)}) \tag{8}$$

where π is a permutation to be determined to minimize $H(\pi)$. This is a typical case in weighted bipartite matching problem, which can be solved in $O(N^3)$ time using the Hungarian algorithm (Papadimitriou and Stieglitz, 1982).
Minimization of $H(\pi)$ over π gives the correspondences at the sample points. The correspondence is extended to the complete shape using the regularized thin plate splines as the aligning transform. Aligning shapes leads to a general measure of shape similarity. The dissimilarity between two shapes can thus be computed as the sum of matching errors between corresponding points. Given this dissimilarity measure, Belongie et al. (2002), apply nearest-neighbor algorithms for object recognition.

4. Integration of local descriptors from multiple views

Depending on how the model of a given object is built, the approaches of using local invariant regions for object recognition can be split into two categories. One takes a single view of the object for developing the model, while the other uses multiple views. Both recognize the object in different views along with occlusions and different geometric and photometric conditions. Because of multiple views of the object considered in the modeling phase, the multi-view based methods can recognize the object in a much broader range of conditions. As far as stereo vision for 3D object recognition is concerned, only the methods using multi-views are considered in this section. Two methods are reviewed, one is given by Lowe (2001) that fuses the SIFT features from multiple views of an object into a single model with view-dependent clusters, and the other, proposed by Rothganger et al. ((2006), builds a patch-based 3D model using affine region descriptors and multi-view spatial constraints.

4.1 Fusion of SIFT features from multiple views

Lowe (2001) proposes a method that combines SIFT features from multiple views to model the appearance of an object for full 3D object recognition. The feature combinations are performed according to the closeness of the geometric fit to existing views, and similar views are fused into view clusters. For nearby views that are not combined, matching features are linked across the views so that a match in one view is automatically propagated as a potential match in neighboring views. Therefore additional training images continue to contribute to the robustness of the model by capturing more feature variation without leading to a continuous increase in the number of view clusters.

Assuming a model of an object built on the SIFT features extracted from a given view of the object, the determination on whether to cluster a new view with the existing model depends on e, which is an error between the model-based projected features and the image features in the new view (Lowe, 2001).

$$e = \sqrt{\frac{2||\mathbf{Ax} - \mathbf{b}||}{r - 4}} \tag{9}$$

where $\mathbf{A}$ is a matrix formed by the coordinates of model-based projected features, r is the number of rows in $\mathbf{A}$, $\mathbf{x}$ is the parameters of the similarity transform (Lowe, 2001), and $\mathbf{b}$ is the the coordinates of image features. Lowe chooses a threshold, T_e, as 0.05 times the maximum dimension of the training image, which results in clustering views that differ by less than roughly 20 degrees rotation in depth. As each new training image arrives, it is matched to the existing model views. Three possible cases can occur:

1. The training image does not match any existing model. In this case, the image is used to form a new model.
2. The training image matches an existing model view, and $e > T_e$. In this case, a new model view is formed from this training image. This is similar to forming a new object model, except that all matching features are linked between the current view and the three closest matching model views.
3. The training image matches an existing model view, and $e \leq T_e$, which means the new training image is to be combined with the existing model view. All features from the new training image are transformed into the coordinates of the model-based view using the similarity transform. The new features are added to those of the existing model view and

linked to any matching features. Any features that are very similar to existing ones (have a distance that is less than a third that of the closest non-matching feature) will be removed, as they do not add significant new information.

The result is that training images that are closely matched by the similarity transform are clustered into model views that combine their features for improved robustness. Otherwise, the training images form new views in which features are linked to their neighbors.
Although Lowe (2001) shows an examples in which a few objects are successfully identified in a cluttered scene, no results are reported on recognizing objects with large viewpoint variations, significant occlusions and illumination variations.

4.2 Patch-based 3D model with affine detector and spatial constraint

Generic 3D objects often have non-flat surfaces. To model and recognize a 3D object given a pair of stereo images, Rothganger et al. (2006 proposes a method for capturing the non-flat surfaces of the 3D object by a large set of sufficiently small patches, their geometric and photometric invariants, and their 3D spatial constraints. Different views of the object can be matched by checking whether groups of potential correspondences found by correlation are geometrically consistent. This strategy is used in the object modeling phase, where matches found in pairs of successive images of the object are used to create a 3D affine model. Given such a model consisting of a large set of affine patches, the object in a test image can be claimed recognized if the matches between the affine regions on the model and those found in the test image are consistent with *local appearance models* and *geometric constraints*. Their approach consists of three major modules:

1. Appearance-based selection of possible matches: Using the Harris affine detector (Section 2) and a DoG-based (Difference-of-Gaussians) interest point detector, corner-like and blob-like affine regions can be detected. Each detected affine region has an elliptical shape. The dominant gradient orientation of the region (Lowe, 2004) can transform an ellipse into a parallelogram and a unit circle into a square. Therefore, the output of this detection process is a set of image regions in the shape of parallelograms. The affine rectifying transformations can map each parallelogram onto a "unit" square centered at the origin, known as a rectified affine region. Each rectified affine region is a normalized representation of the local surface appearance, invariant to planar affine transformations. The rectified affine regions are matched across images of different views, and those with high similarity in appearance are selected as an initial match set to reduce the cost of latter constrained search. An example of the matched patch pairs on a teddy bear, reproduced from Rothganger et al. (2006, is shown in Fig. 7
2. Refine selection using geometrical constraints: RANSAC (RANdom SAmple Consensus, Fischler & Bolles 1981) is applied to the initial appearance-based matched set to find a geometrically consistent subset. This is an iterative process that keeps on until a sufficiently large geometrically consistent set is found, and the geometric parameters are finally renewed. The patch pairs which appear to be similar in Step 1 but fail to be geometrically consistent are removed in this step.
3. Addition of geometrically consistent matches: Explore the remainder of the space of all matches, and search for other matches which are consistent with the established geometric relationship between the two sets of patches. Obtaining a nearly maximal set of matches can improve recognition, where the number of matches acts as a confidence measure, and object modeling, where they cover more surface of the object.

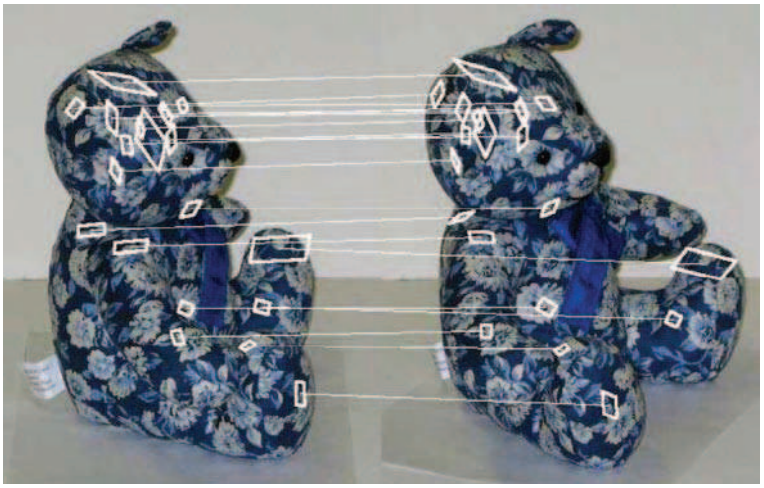


Fig. 7. An example of the matched patches between two images, reproduced from Rothganger et al. ((2006).

To verify their proposed approach, Rothganger et al. (2006) design an experiment that allows an object's model built on tens of images taken from cameras roughly placed in an equatorial ring centered at the object. Fig. 8 shows one such training set, composed of images used in building the model for the object "teddy bear". Fig. 9 shows all the objects with models built from the patches extracted from the training sets. Table 1 summarizes the number of images in the training set of each object, along with the number of patches extracted from each training set for forming the object's model. The model is evaluated in recognizing the object in cluttered scenes with it placed in arbitrary poses and, in some cases, partial occlusions. Fig. 10 shows most test images for performance evaluation. The outcomes of this performance evaluation, among others, will be presented in the next section.

	Apple	Bear	Rubble	Salt	Shoe	Spidey	Truck	Vase
Training images	29	20	16	16	16	16	16	20
Model patches	759	4014	737	866	488	526	518	1085

Table 1. Numbers of training images and patches used in the model for each object in the object gallery shown in Fig. 9

5. Performance evaluation and benchmark databases

As reviewed in Section 4, only few methods develop object recognition models on interest points with information integrated across stereo or multiple views; however, many build their models with one single image or a set of images without considering the 3D geometry of the objects. The view-clustering method by Lowe (2001), reviewed in Section 4.1, can be considered in between of these two categories. Probably because few works of the same category are available, Lowe (2001) does not present any comparison with other methods using multiple views. Nevertheless, Rothganger et al. ((2006) report a performance comparison of their method with a few state-of-the-art algorithms using the training and test images as shown in Fig.10. This comparison study is briefly reviewed below, followed by an introduction to the databases that offer samples taken in stereo or multiple views.

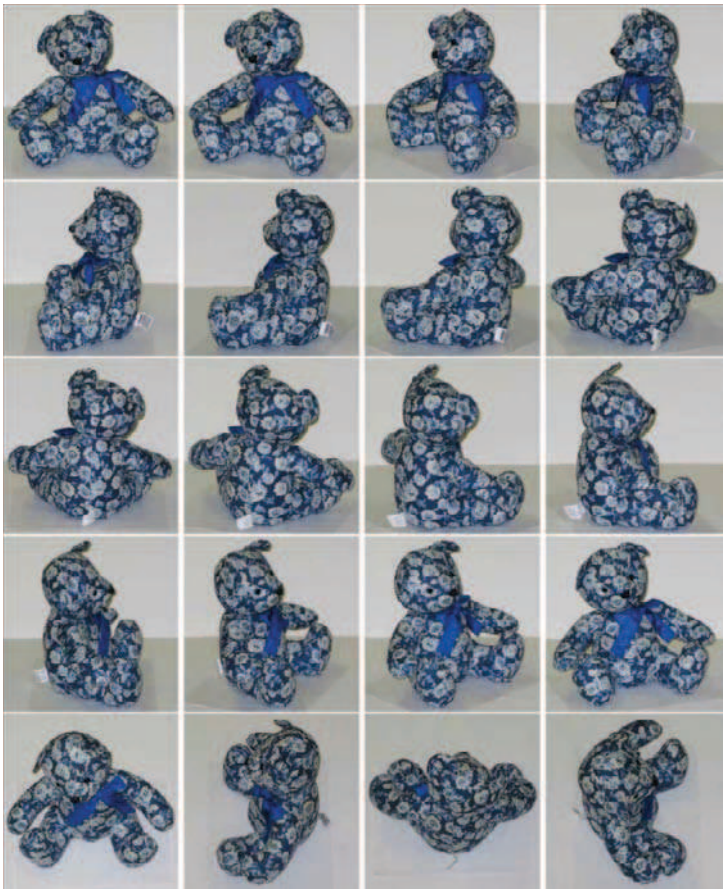


Fig. 8. The training set used in building the model for ”teddy bear”, reproduced from Rothganger et al. ((2006).

5.1 Performance comparison in a case study

This section summarizes the performance comparison conducted by Rothganger et al. ((2006), which include the algorithms given by Ferrari et al. (2004), Lowe (2004), Mahamud & Hebert (2003), and Moreels et al. (2004). The method by Lowe (2004) has been presented in Section 3, and the rest are addressed below.

Mahamud & Hebert (2003) develop a multi-class object detection framework with a nearest neighbor (NN) classifier as its core. They derive the optimal distance measure that minimizes a nearest neighbor mis-classification risk, and present a simple linear logistic model which measures the optimal distance in terms of simple features like histograms of color, shape and texture. In order to perform search over large training sets efficiently, their framework is extended to finding the Hamming distance measures associated with simple discriminators. By combining different distance measures, a hierarchical distance model is constructed, and their complete object detection system is an integration of the NN search over object part classes.

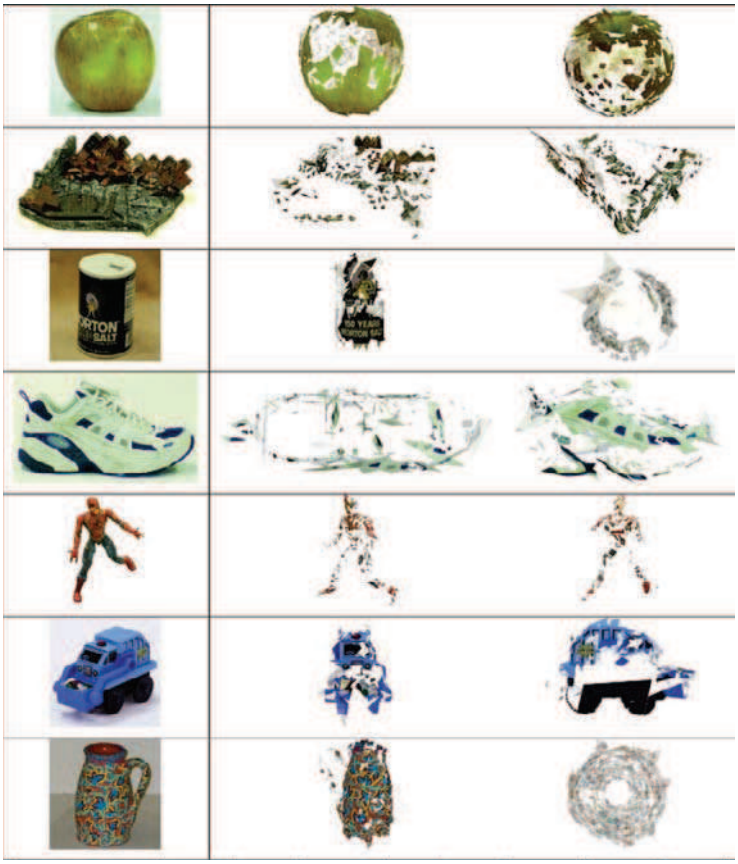


Fig. 9. Object gallery. Left column: One of several input pictures for each object. Right column: Renderings of each model, not necessarily in same pose as input picture, reproduced from Rothganger et al. ((2006).

The method proposed by Ferrari et al. (2004) is initialized by a large set of unreliable region correspondences generated purposely to maximize the amount of correct matches, at the cost of producing many mismatches. A grid of circular regions is generated for covering the modeling image[1]. The method then iteratively alternates between expansion and contraction phases. The former aims at constructing correspondences for the coverage regions, while the latter attempts to remove mismatches. At each iteration, the newly constructed matches between the modeling and test images help a filter to take better mismatch removal decisions. In turn, the new set of supporting regions makes the next expansion more effective. As a result, the amount, and the percentage, of correct matches grows every iteration.
Moreels et al. (2004) proposes a probabilistic framework for recognizing objects in images of cluttered scenes. Each object is modeled by the appearance of a set of features extracted from a single training image, along with the position of the feature set with respect to a common

[1] *Modeling* images or *training* images refer to the image samples used in building an object's model.

Fig. 10. The test set for performance evaluation, the objects shown in Fig. 1 are placed in arbitrary poses in cluttered scenes and, in some cases, with partial occlusions; reproduced from Rothganger et al. ((2006).

reference frame. In the recognition phase, the object and its position is estimated by finding the best interpretation of the scene in terms of object models. Features detected in a test image are hypothesized as features from either the database or clutters. Each hypothesis is scored using a generative model of the image which is defined using the object models and a model for clutter. Heuristics are explored to find the best from a large hypothesis space, improving the performance of this framework.

As shown in Fig. 11, Rothganger et al.'s and Lowe's algorithms perform best with true positive rates over 93% at false positive rate 1%. The algorithm by Ferrari et al. keeps improving its performance as the false positive rate is allowed to increase, and can reach $> 95\%$ in true positive rate if the false positive rate increases to 7.5%. It is interesting to see that two of Rothganger et al.'s methods (color and black-and-while) and Lowe's method perform almost equally well across for all false positive rates shown. This can be caused by the fact that their models can fit to the objects in most views, but fail in a few specific views because of the lack of samples from these views used in building the model. Although all tested

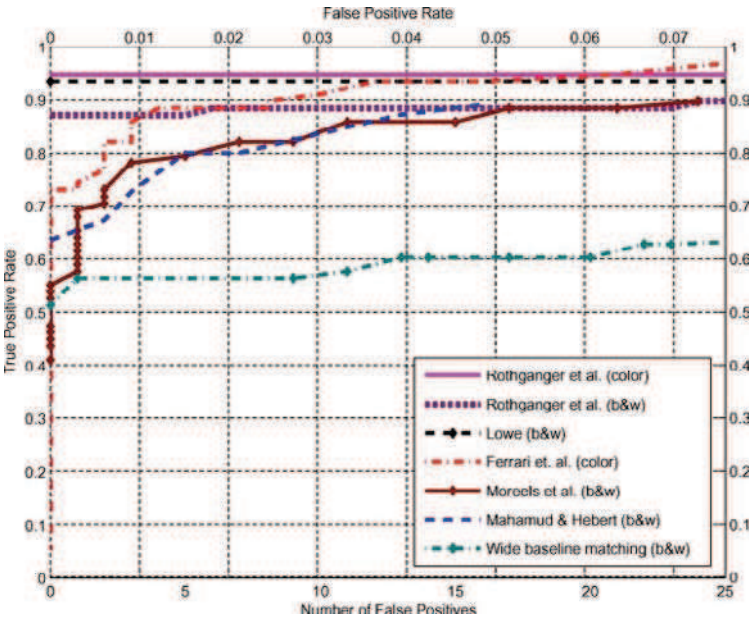


Fig. 11. Performance comparison reported in Rothganger et al. ((2006).

algorithms use multiple views to build object models, only Lowe's and Rothganger et al.'s algorithms combine the information from across multiple views for recognition. The rest consider all modeling images independently, without looking into geometric relationships between these images, and tackle object recognition as an image match problem. To evaluate the contribution made from geometric relationships, Rothganger et al. ((2006) have studied a base line recognition method where the pairwise image matching part of their modeling algorithm is used as the recognition kernel. An object is considered recognized when a sufficient percentage of the patches found in a training image are matched to the test image. The result is shown in Fig. 11 in the green doted line, it performs worst in all range of false positive rates.

5.2 Databases for 3D object recognition

The database used in Rothganger et al. ((2006) consists of 9 objects and 80 test images. The training images are stereo views for each of the 9 objects that are roughly equally spaced around the equatorial ring for each of them, as an example "teddy bear" shown in Fig. 8. The number of stereo views ranges from 7 to 12 for different objects. The test images, shown in Fig. 10, are monocular images of objects under varying amounts of clutter and occlusion and different lighting conditions. It can be downloaded at `http://www-cvr.ai.uiuc.edu/~kushal/Projects/StereoRecogDataset/`. In addition, several other databases can also be considered for benchmarking stereo vision algorithms for object recognition. The ideal databases must offer stereo images for training, and test images collected with variations in viewpoint, scale, illumination, and partial occlusion.

Columbia Object Image Library (COIL-100) database offers 7,200 images of 100 objects (72 images per object). The objects have a wide variety of complex geometric and reflectance characteristics. The images were taken under well-controlled conditions. Each object was placed on a turntable, and an image was taken by a fixed camera when the turntable made a 5^o rotation. Most studies take a subset of images with viewing angles equally apart for training, and the rest for testing. A few samples are shown in Fig. 12. It serves as a good database for evaluating object recognition with viewpoint variation, but is inappropriate for testing against other variables. COIL-100 can be downloaded via `http://www1.cs.columbia.edu/CAVE/software/softlib/coil-100.php`.

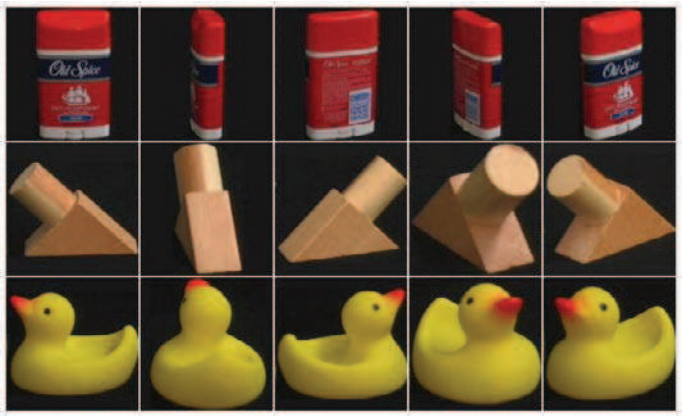


Fig. 12. Samples from COIL-100.

The Amsterdam Library of Object Images (ALOI), made by Geusebroek et al. (2005), offers 1,000 objects with images taken under various imaging conditions. The primary variables considered include 72 different viewing angles with 5^o apart, 24 different illumination conditions, and 12 different illumination colors in terms of color temperatures. 750 out of the 1,000 objects were also captured with wide baseline stereo images. Figs. 13, 14, and 15 give samples in viewpoint change, illumination variation, and stereo, respectively. The stereo images can be used for training, and the rest can be used for testing. This dataset appears better than COIL-100 in terms of offering samples of a large amount of objects with a broader scope of variables. ALOI can be downloaded via `http://staff.science.uva.nl/~aloi/`.

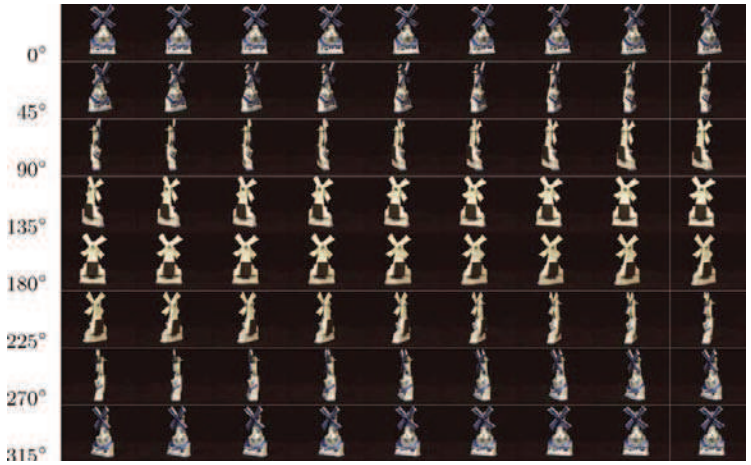


Fig. 13. A example viewpoint subset from ALOI database, reproduced from Geusebroek et al. (2005).

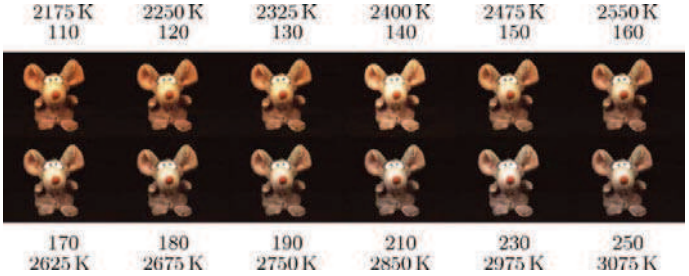


Fig. 14. A example of illumination subset from ALOI database, reproduced from Geusebroek et al. (2005).

The ETHZ Toy database offers 9 objects with single or multiple views for modeling, and 23 test images with different viewpoints, scales, and occlusions in cluttered backgrounds. Fig. 16 shows 2 sample objects and each with 5 training images, and Fig. 17 shows 15 out of the 23 test images. It can be downloaded via `http://www.vision.ee.ethz.ch/~calvin/datasets.html`.

6. Conclusion

This chapter discusses methods using affine invariant descriptors extracted from stereo or multiple training images for object recognition. It focuses on the few that integrate information from multiple views in the model development phase. Although the objects in single test images can appear in different viewpoint, scale, illumination, blur, occlusion, and image quality, the training images must be taken from multiple views, and thus can only have different viewpoints and probably a little scale variation.

Because of their superb invariance to viewpoint and scale changes, Hessian-Affine, Harris-Affine, and MSER detectors are introduced as the most appropriate ones for extracting

Fig. 15. A sample stereo subset from ALOI database, reproduced from Geusebroek et al. (2005).

Fig. 16. Sample training images of 2 objects from the ETHZ Toys database.

Fig. 17. 15 sample test images from the ETHZ Toys database.

interest regions from the training set. SIFT and shape context are selected as two promising descriptors for representing the extracted interest regions. Methods that combine the aforementioned affine detectors and descriptors for 3D object recognition are yet to develop, but the view-clustering in Lowe (2001) and the modeling with geometric consistency in Rothganger et al. ((2006) serve as good references for integrating information from multiple views. A sample performance evaluation study is introduced along with several benchmark databases that offer stereo or multiple views for training. This chapter is expected to offer some perspectives toward potential research directions in the stereo correspondence with local descriptors for 3D object recognition.

7. Acknowledgement

This research is supported by Taiwan National Science Council (NSC) under grant 99-2221-E-011-098.

8. References

Belongie, S., Malik, J. & Puzicha, J. (2002). Shape matching and object recognition using shape contexts, 24(4): 509–522.

Ferrari, V., Tuytelaars, T. & Gool, L. J. V. (2004). Simultaneous object recognition and segmentation by image exploration, *ECCV (1)*, pp. 40–54.

Fischler, M. A. & Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, *Commun. ACM* 24(6): 381–395.

Forssén, P.-E. & Lowe, D. G. (2007). Shape descriptors for maximally stable extremal regions, *ICCV*, pp. 1–8.

Freeman, W. T. & Adelson, E. H. (1991). The design and use of steerable filters, *IEEE Trans. Pattern Anal. Mach. Intell.* 13(9): 891–906.

Geusebroek, J.-M., Burghouts, G. J. & Smeulders, A. W. M. (2005). The amsterdam library of object images, *International Journal of Computer Vision* 61(1): 103–112.

Gool, L. J. V., Moons, T. & Ungureanu, D. (1996). Affine/ photometric invariants for planar intensity patterns, *ECCV (1)*, pp. 642–651.

Ke, Y. & Sukthankar, R. (2004). Pca-sift: a more distinctive representation for local image descriptors, *CVPR*, pp. 506–513.

Koenderink, J. J. & van Doom, A. J. (1987). Representation of local geometry in the visual system, *Biol. Cybern.* 55(6): 367–375.

Lazebnik, S., Schmid, C. & Ponce, J. (2003). A sparse texture representation using affine-invariant regions, *CVPR (2)*, pp. 319–326.

Lindeberg, T. (1998). Feature detection with automatic scale selection, *International Journal of Computer Vision* 30(2): 79–116.

Lindeberg, T. & Gårding, J. (1997). Shape-adapted smoothing in estimation of 3-d shape cues from affine deformations of local 2-d brightness structure, *Image Vision Comput.* 15(6): 415–434.

Lowe, D. G. (1999). Object recognition from local scale-invariant features, *ICCV*, pp. 1150–1157.

Lowe, D. G. (2001). Local feature view clustering for 3d object recognition, *CVPR (1)*, pp. 682–688.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision* 60(2): 91–110.

Mahamud, S. & Hebert, M. (2003). The optimal distance measure for object detection, *CVPR (1)*, pp. 248–258.

Matas, J., Chum, O., Urban, M. & Pajdla, T. (2002). Robust wide baseline stereo from maximally stable extremal, *In British Machine Vision Conference*, pp. 384–393.

Mikolajczyk, K. & Schmid, C. (2001). Indexing based on scale invariant interest points, *ICCV*, pp. 525–531.

Mikolajczyk, K. & Schmid, C. (2002). An affine invariant interest point detector, *ECCV (1)*, pp. 128–142.

Mikolajczyk, K. & Schmid, C. (2004). Scale & affine invariant interest point detectors,

International Journal of Computer Vision 60(1): 63–86.

Mikolajczyk, K. & Schmid, C. (2005). A performance evaluation of local descriptors, *IEEE Trans. Pattern Anal. Mach. Intell.* 27(10): 1615–1630.

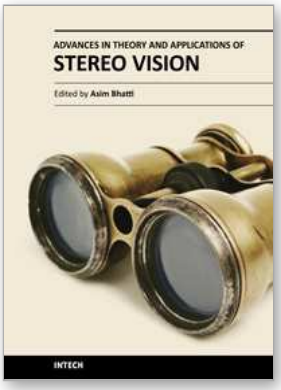
Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T. & Gool, L. J. V. (2005). A comparison of affine region detectors, *International Journal of Computer Vision* 65(1-2): 43–72.

Moreels, P., Maire, M. & Perona, P. (2004). Recognition by probabilistic hypothesis construction, *ECCV (1)*, pp. 55–68.

Nistér, D. & Stewénius, H. (2008). Linear time maximally stable extremal regions, *ECCV (2)*, pp. 183–196.

Rothganger, F., Lazebnik, S., Schmid, C. & Ponce, J. ((2006)). 3d object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints, *International Journal of Computer Vision* 66(3): 231–259.

Schaffalitzky, F. & Zisserman, A. (2002). Multi-view matching for unordered image sets, or ”how do i organize my holiday snaps?”, *ECCV (1)*, pp. 414–431.



Advances in Theory and Applications of Stereo Vision
Edited by Dr Asim Bhatti

ISBN 978-953-307-516-7
Hard cover, 352 pages
Publisher InTech
Published online 08, January, 2011
Published in print edition January, 2011

The book presents a wide range of innovative research ideas and current trends in stereo vision. The topics covered in this book encapsulate research trends from fundamental theoretical aspects of robust stereo correspondence estimation to the establishment of novel and robust algorithms as well as applications in a wide range of disciplines. Particularly interesting theoretical trends presented in this book involve the exploitation of the evolutionary approach, wavelets and multiwavelet theories, Markov random fields and fuzzy sets in addressing the correspondence estimation problem. Novel algorithms utilizing inspiration from biological systems (such as the silicon retina imager and fish eye) and nature (through the exploitation of the refractive index of liquids) make this book an interesting compilation of current research ideas.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Gee-Sern Jison Hsu (2011). Stereo Correspondence with Local Descriptors for Object Recognition, Advances in Theory and Applications of Stereo Vision, Dr Asim Bhatti (Ed.), ISBN: 978-953-307-516-7, InTech, Available from: http://www.intechopen.com/books/advances-in-theory-and-applications-of-stereo-vision/stereo-correspondence-with-local-descriptors-for-object-recognition

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821