CHAPTER 1.5

PATTERN RECOGNITION WITH LOCAL INVARIANT FEATURES

C. Schmid[1], G. Dorkó[1], S. Lazebnik[2], K. Mikolajczyk[1] and J. Ponce[2]
[1] *INRIA Rhône-Alpes, GRAVIR-CNRS*
655, av. de l'Europe, 38330 Montbonnot, France
[2] *Dept. of Computer Science and Beckman Institute*
University of Illinois, Urbana, IL 61801, USA

Local invariant features have shown to be very successful for recognition. They are robust to occlusion and clutter, distinctive as well as invariant to image transformations. In this chapter recent progress on local invariant features is summarized. It is explained how to extract scale and affine-invariant regions and how to obtain discriminant descriptors for these regions. It is then demonstrated that combining local features with pattern classification techniques allows for texture and category-level object recognition in the presence of varying viewpoints and background clutter.

1. Introduction

Local photometric invariants have become more and more popular over the past years. This is due to (i) their locality which permits recognition in the presence of occlusion and clutter, (ii) their distinctiveness due to the use of photometric information and (iii) their invariance which makes them stable under image transformations and illumination changes. This is in contrast to classical recognition approaches which were mostly based on global photometric information or geometric features.

Color histograms [37] and eigenimages [38] are examples of recognition methods based on global photometric information. They require segmentation in the presence of clutter and are not robust to occlusions. Furthermore, they are not invariant to image transformations. However, these methods can discriminate due to the use of photometric information and they perform very well in constrained settings. Recognition based on geometric features, on the other hand, does not require segmentation and is invariant to image transformations [30]. However, the features are not very discriminating and are sensitive to noise and errors in the extraction process.

The idea of combining the distinctiveness of photometric information with the locality and invariance of geometric features has led to the development of local pho-

tometric invariants. An initial solution was presented by Schmid and Mohr [35]. They describe the image with a set of rotation-invariant descriptors computed at automatically extracted interest points. A multi-scale framework makes the description invariant to similarity transformations. This description combined with a voting scheme and neighbourhood constraints allows for excellent recognition results. The approach is, however, limited to similarity transformations and only images of the same object or scene can be recognized. A solution is possible due to the following extensions: (i) the extraction of affine-invariant photometric descriptors and (ii) the use of pattern classification techniques. Section 2 describes the extraction of local invariant features and their application to recognition in the presence of viewpoint changes. Section 3 and Section 4 demonstrate that combining local features with pattern classification techniques allows for texture and category-level object recognition in the presence of varying viewpoints and background clutter.

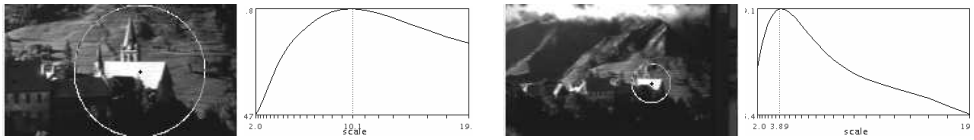
2. Extraction of local invariant features

This section presents the extraction of local invariant features. Sections 2.1 and 2.2 present scale and affine-invariant detectors and in Section 2.3 different descriptors are compared. An application to recognition is presented in Section 2.4.

2.1. *Scale-invariant regions*

Most scale-invariant detectors search for maxima in the 3D scale-space representation of an image (x, y and $scale$). They differ mainly in the differential expression used to build the scale-space representation. Crowley [6] detects local features in an image pyramid. Lindeberg [16] searches for 3D maxima of the scale-normalized Laplacian-of-Gaussian (LoG). The LoG operator detects blob-like structures. Lowe [19] uses the difference-of-Gaussian (DoG) to approximate the LoG. This results in an efficient algorithm for extracting local 3D extrema in scale-space. Kadir and Brady [11] use local complexity as saliency measure. They compute the entropy of greylevel intensities over a neighborhood and then search for entropy maxima in scale and location.

Our approach, the Harris-Laplace detector [21,24], selects a complementary type of regions: corners and regions of "high information content" (cf. Figure 9). The detector first extracts interest points at multiple scale levels which are the local spatial maxima of the scale-adapted Harris function. It then uses the Laplacian for scale selection [16]. Scale selection determines the *characteristic* scale of a local structure, i.e. the scale with maximum similarity between the feature detection operator and the local image structures. This scale is determined as the extremum over scale of a given function (see Figure 1) and is independent of the image resolution. Note that there might be several extrema, that is several characteristic scales corresponding to different local structures centered at one point. The Laplacian operator is used for scale selection since it gave the best results in our experimental comparison [21].



(a) (b) (c) (d)

Fig. 1. Example of characteristic scales. (a),(c) images with different resolutions; the radius of displayed regions is equal to 3 times the characteristic scale. (b),(d) responses of the normalized LoG over scales for (a),(c). The characteristic scales are 10.1 and 3.89 for (a) and (c), respectively. The ratio of scales corresponds to the scale factor (2.5) between the two images.

Initial multi-scale interest points are rejected if the Laplacian attains no extremum at the scale of extraction or if the Laplacian response is below a given threshold. We then obtain a set of scale-invariant points with associated scales. Figure 2 presents an example of points detected with Harris-Laplace. Images (a) and (b) show points detected with the multi-scale Harris detector; the radius of displayed regions equals 3 times the detection scale. Points corresponding to the same local structure are selected manually. Note that interest points, detected for the same image structure, change their location relative to the detection scale in the gradient direction. Images (c) and (d) show the points selected by the Laplacian. Note that two or more points can be selected, as several local maxima might exist in scale-space. We can see that the location and the scale of the points correspond to the transformation between the images.

(a) (b) (c) (d)

Fig. 2. Scale-invariant interest point detection. (a), (b) Initial multi-scale Harris points corresponding to one local structure (selected manually). (c), (d) Scale-invariant interest points selected with the Harris-Laplace detector.

An experimental evaluation of the Harris-Laplace detector for images of real scenes [21,24] has shown a very good performance up to a scale factor of 4. The performance was measured by the repeatability rate, that is the percentage of points detected at the same relative position and with corresponding regions, as well as by the performance in the context of recognition.

2.2. *Affine-invariant regions*

The Harris-Laplace detector is not invariant to significant affine transformations. Figure 3 shows Harris-Laplace regions in black and the corresponding regions pro-

jected with the affine transformation to the other image in white. The regions detected with Harris-Laplace do not cover the same part of the affine transformed image. Affine-invariant region extraction is therefore required.

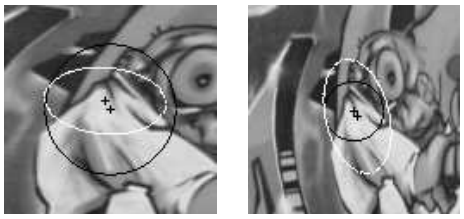


Fig. 3. Scale-invariant interest point detection for an image pair related by an affine transformation. The regions detected with Harris-Laplace in black and the corresponding regions projected with the affine transformation to the other image in white.

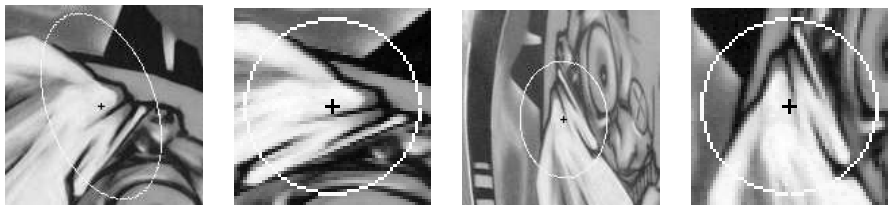
Alvarez and Morales [2] proposed an affine-invariant algorithm for corner detection which uses affine morphological multi-scale analysis. Their approach only applies to perfect corners. Lindeberg and Gårding [17] presented a method for finding blob-like affine features. Their approach first extracts maxima of the normalized Laplacian in scale-space and then modifies iteratively the scale and local affine shape of the regions. Local affine shape is determined with the second moment matrix. Baumberg [4] as well as Schaffalitzky and Zisserman [33] use the local affine shape estimation to adapt the point neighbourhood of multi-scale Harris points and Harris-Laplace points respectively. Tuytelaars and Van Gool [39] combine Harris points with nearby edges. An affine-invariant parallelogram region is determined by a Harris point and one point on each of two nearby edges. They also proposed a purely intensity-based method. It is initialized with local intensity extrema. For each extremum the algorithm finds significant changes in the intensity profiles along rays going out from the extremum. An ellipse is fitted to the region defined by the locations of these changes. Matas et al. [20] introduced maximally stable extremal regions (MSERs). An extremal region is a connected component of pixels which are all brighter or darker than all pixels on the region's boundary. These regions are invariant to affine transformations as well as to monotonic intensity transformations.

Our approach, the Harris-Affine detector [22,24] extends the Harris-Laplace detector by estimating the local affine shape based on the second moment matrix [17]. The second moment matrix describes the local image structure and is defined by the gradient distribution in a point neighbourhood:

$$\mu(\mathbf{x}, \sigma_I, \sigma_D) = \sigma_D^2 \, g(\sigma_I) * \begin{bmatrix} L_x^2(\mathbf{x}, \sigma_D) & L_x L_y(\mathbf{x}, \sigma_D) \\ L_x L_y(\mathbf{x}, \sigma_D) & L_y^2(\mathbf{x}, \sigma_D) \end{bmatrix}$$

The local derivatives are computed by convolution with Gaussian derivatives of scale σ_D (differentiation scale). The derivatives are then averaged in the point neighborhood by smoothing with a Gaussian of scale σ_I (integration scale). The

eigenvalues of the second moment matrix determine the affine shape of the point neighbourhood. Affine normalization projects the affine pattern to the one with equal eigenvalues, i.e. uses as transformation the square root of the second moment matrix. See Figure 4 for illustration. The normalized regions are isotropic in terms of the second moment matrix and are related by a simple rotation. Rotation preserves the eigenvalue ratio for an image patch and the affine deformation can therefore be determined up to a rotation factor.



(a) (b) (c) (d)

Fig. 4. Affine normalization with the second moment matrix: (a), (c) initial images with the affine shape matrices. (b), (d) normalized images. The normalized images are related by a rotation.

In practice the second moment matrix as well as the characteristic scale change if the patch is transformed and therefore need to be re-estimated iteratively. Initial points are obtained by the multi-scale Harris detector. For each point we iteratively estimates location, scale and local shape. Figure 5 shows points and regions detected in consecutive steps of the iterative procedure. For this example, the location, scale and shape of the point do not change after four iterations. We stop iterating when the second moment matrix μ (of the transformed patch) is sufficiently close to a pure rotation. This is measured by the similarity of the eigenvalues $\lambda_{max}(\mu)$ and $\lambda_{min}(\mu)$, i.e. $\frac{\lambda_{min}(\mu)}{\lambda_{max}(\mu)} > 0.95$ for our experiments. Another important point is to stop in the case of divergence. In theory there is a singular case when the eigenvalue ratio tends to infinity, i.e. on a step-edge. Therefore, the point should be rejected if the ratio of the eigenvalues is too large (i.e. bigger than 6), otherwise it leads to unstable elongated structures. The convergence properties of the shape adaptation algorithm are extensively studied in [17]. It is shown that except for the singular case the point of convergence is always unique. In general the procedure converges to the correct solution provided that the initial estimate of the affine deformation is sufficiently close to the true deformation, and the scale is correctly selected with respect to the size of the local image structure.

Figure 6 presents an example of points detected with Harris-Affine. Images (a) and (b) show the multi-scale Harris points used for initialization, in black the region selected by the Harris-Laplace detector. Images (c) and (d) show the points and regions after an iterative estimation for each point in (a) and (b). Note that the points in (a) and (b) which correspond to the same physical structure, but are detected at different locations due to scale, converge to the same point location and

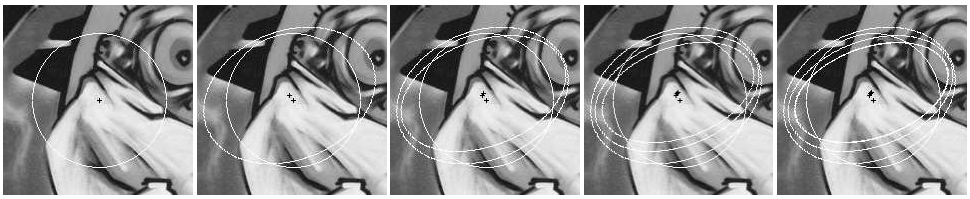


Fig. 5. Iterative detection of an affine-invariant interest point. The left image shows the point used for initialization. The consecutive images show the points and regions after iterations 1, 2, 3 and 4.

region. We can see that the method converges correctly even if the location and the scale of the initial point is relatively far from the point of convergence. It is straightforward to identify these points by comparing their location, scale and local shape and to select one of them.

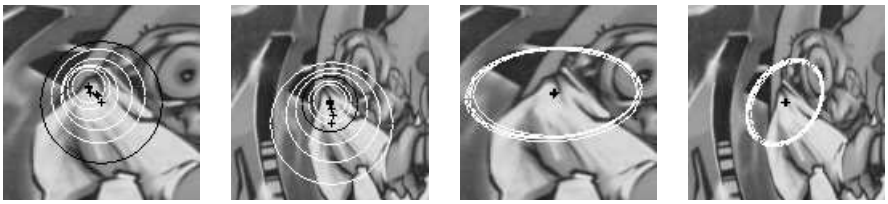


Fig. 6. Affine-invariant interest point detection : (a), (b) Multi-scale Harris points/regions (in black – Harris-Laplace). The radius of the circles is three times the detection scale. (c),(d) Points and affine regions obtained with the iterative algorithm applied to points in (a) and (b). Note that points representing the same structure converge to the same solution.

An experimental evaluation of the Harris-Affine detector for images of real scenes [22,24] has shown a very good performance up to a viewpoint change of 70 degrees. Harris-Laplace shows a similar performance up to 30 degrees of viewpoint change. For larger viewpoint changes the performance of Harris-Laplace decreases rapidly.

2.3. *Descriptors*

To obtain local invariant features, the extracted and normalized regions have to be described. Many different image descriptors have been developed. They should be distinctive and at the same time robust to viewpoint changes as well as to errors of the detector. In the following we have selected a few promising descriptors and compare them in the context of recognition. Note that the normalized regions are not rotation-invariant. To eliminate rotation our comparison uses rotation-invariant descriptors. A simple way to obtain rotation invariance is to orientate the patch in the direction of the dominant gradient.

The compared descriptors are presented in the following. SIFT descriptors [19] describe the gradient distribution in a region by a 3D histogram of location and gradient orientation. The quantization of location and gradient orientation makes the descriptor robust to small geometric distortions and small errors in the region extraction. Steerable filters [10] steer derivatives in a particular direction. Steering in direction of the gradient orientation makes them invariant to rotation. Differential invariants [9] are combinations of image derivatives which are rotation-invariant. Complex filters [33] differ from the Gaussian derivatives by a linear coordinate change in filter response space. Moment invariants [40] characterize the shape and the intensity distribution in a region. Cross-correlation for a sub-sampled image patch is used as a baseline descriptor.

Our evaluation criterion is the ROC (receiver operating characteristics) of the detection rate for the query image with respect to the false positive rate in a database of images. Detection rate is the number of correctly matched points with respect to the number of possible correct matches. False positive rate is the probability of a false match in the database of descriptors, that is the total number of false matches with respect to the product of the number of database points and the number of query image points. The similarity threshold between descriptors is varied to obtain the ROC curves. The results are displayed for a false positive rate up to 0.012, that is each point from the query image matches at most with 1.2% of points in the database. The threshold is usually set below this value; otherwise the number of false matches is too high to allow reliable recognition.

Figure 7 compares the performance in the presence of an affine transformation between image pairs for which the viewpoint of the camera is changed by 60 degrees. This introduces a perspective transformation which can be locally approximated by an affine transformation. There are also some scale and brightness changes in the test images. To eliminate the effects of the affine transformation, we use the Harris-Affine detector. The descriptors are computed on point neighborhoods normalized with the locally estimated affine transformations. SIFT descriptors perform better that the other ones and steerable filters come second, but they perform significantly worse than SIFT. Note that SIFT descriptors computed on Harris-Laplace regions perform worse than any of the other descriptors (see `HL sift`), as these regions and therefore the descriptors are only scale and not affine-invariant. Cross-correlation obtains the lowest score. This can be explained by the sensitivity of cross-correlation to errors in the point and region extraction.

A comparison for image rotation, scale and illumination changes [23] shows similar results as in the case of affine viewpoint changes. In conclusion we observe that the performance varies significantly for different descriptors. SIFT descriptors perform best. This shows the robustness and the distinctiveness of the region-based SIFT descriptor. Second best are steerable filters; they can be considered a good choice given their low dimensionality. Cross-correlation gives unstable results. The performance depends on the accuracy of interest point and region detection, which

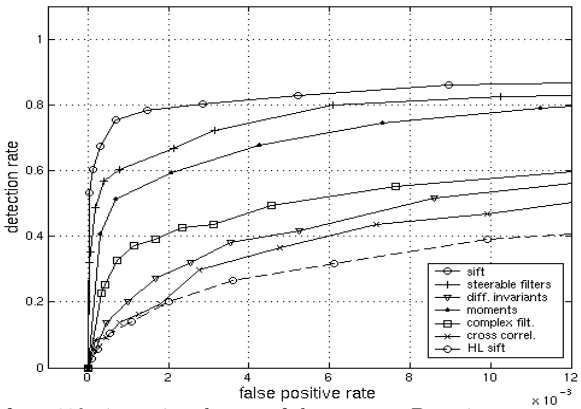


Fig. 7. Evaluation for a 60° viewpoint change of the camera. Descriptors are computed for Harris-Affine regions. `HL sift` is the SIFT descriptor computed for Harris-Laplace regions.

decreases for significant geometric transformations. The differential invariants give significantly worse results than the steerable filters, which is surprising as they are based on the same basic components (Gaussian derivatives). The multiplication of derivatives necessary to obtain the rotation invariance increases the instability of the descriptors. Overall, the comparison shows that a robust region-based descriptor performs better than point-wise descriptors.

2.4. *Recognition*

Local invariant photometric descriptors are suitable for recognition of the same object or scene in the presence of large viewpoint changes. Several approaches based on local invariants have been developed [19,22,29,33] and have shown an excellent performance. They all combine similarity measures of local descriptors with semi-local and global consistency constraints.

Our approach [22] extracts affine-invariant regions with the Harris-Affine detector and describes each of them with a set of Gaussian derivatives invariant to rotation and affine illumination changes. The similarity of descriptors is measured by the Mahalanobis distance where the covariance matrix is estimated statistically over a large set of images. A voting algorithm is used to select the most similar images in the database. For each interest point of a query image, its descriptor is compared to the descriptors in the database. If the distance is less than a fixed threshold, a vote is added for the corresponding database image. Note that a point cannot vote several times for the same database image. Votes are then verified by robustly estimating the geometric transformation between image pairs based on RANdom SAmple Consensus (RANSAC) and rejecting inconsistent matches. For our experimental results the transformation is either a homography or a fundamental matrix. A model selection algorithm [12] can be used to automatically decide which transformation is the most appropriate one.

Figure 8 illustrates retrieval results from a database with more than 5000 images. The top row displays query images for which the corresponding image in the database (second row) was correctly retrieved. Note the significant transformations between the query images and the images in the database. There is a scale change of a factor of 3 between images of pair (a). Image pairs (b) and (c) show large viewpoint changes. The displayed matches are the inliers to a robustly estimated fundamental matrix or homography between the query image and the database image.

(a) (b) (c)

Fig. 8. The top row shows the query images and the bottom row shows the most similar images in the database. The displayed matches are the inliers to a robustly estimated fundamental matrix or homography between the query image and the database image. There are (a) 22 matches, (b) 34 matches and (c) 22 matches. All of them are correct.

3. Recognizing textures

We address in this section the problem of representing and recognizing non-rigid textures observed from arbitrary viewpoints. Recent approaches to texture recognition [18,42] perform impressively well on datasets as challenging as the Brodatz database [5]. Unfortunately, these schemes rely on restrictive assumptions about their input (e.g., the texture must be stationary) and are not generally invariant under 2D similarity and affine transformations, much less 3D transformations caused by camera motions and non-rigid deformations of textured surfaces. In addition, most existing approaches to texture analysis use a dense representation where some local image descriptor is computed over a fixed neighborhood of each pixel. Affine-invariant patches can be used to address the issues of *spatial selection*—finding a sparse set of texture descriptors at "interesting" image locations—and *shape*

selection—computing shape and scale characteristics of the descriptors—(see [32] for related work). In addition, they afford a texture representation that is invariant under any geometric transformation that can be *locally* approximated by an affine model: Local affine invariants are capable of modeling not only global affine transformations of the image, but also perspective distortions and non-rigid deformations that preserve the locally flat structure of the surface (e.g., the bending of paper or cloth). In this context, it is appropriate to combine the Harris-Affine interest point detector [22] with the affine-adapted Laplacian blob detector proposed by Lindeberg and Gårding [17]. The two feature detectors are dubbed H (for Harris) and L (for Laplacian), and they provide two description "channels" for local image patterns. Their output on two sample images is shown in Figure 9. Intuitively, the two detectors provide complementary kinds of information: H responds to corners and other regions of "high information content", while L produces a perceptually plausible decomposition of the image into a set of blob-like primitives.

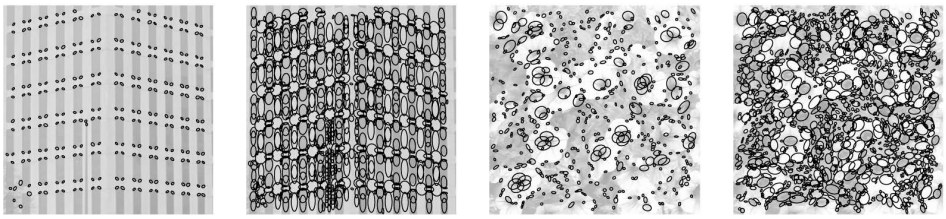


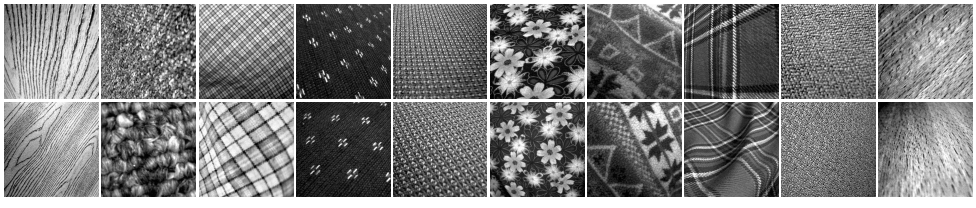
Fig. 9. From left to right: H-detector and L-detector output for a building and flower image.

3.1. *Image signatures*

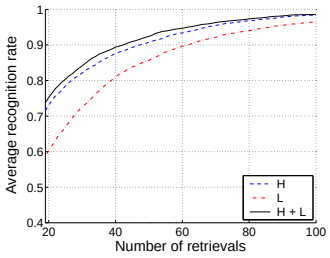
This section demonstrates the power of affine-invariant features as image descriptors in texture classification tasks. Our approach applies the following process to each image in the database using both the H and L feature detectors: (1) find the affine-invariant patches; (2) construct an affine-invariant description of these patches; (3) find the most significant clusters of similar descriptions and use them to construct the *signature* of the image; and (4) compare all pairs of signatures using the Earth Mover's Distance (EMD).

Like most approaches to texture analysis, ours relies on clustering to discover a small set of basic primitives in the initial collection of candidate texture elements. We use a standard agglomerative clustering algorithm. The final representation for the image is a *signature* of the form $\{(m_1, u_1), (m_2, u_2), \ldots, (m_k, u_k)\}$, where m_i is the *medoid* (the most centrally located element of the ith cluster) and u_i is the relative weight of the cluster (the size of the cluster divided by the total number of descriptors extracted from the image). Signatures have been introduced by Rubner *et al.* [31] as representations suitable for matching using the *Earth Mover's Distance* (EMD). For our application, the signature/EMD framework offers several important

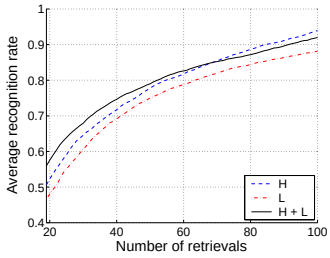
advantages. A signature is more descriptive than a histogram, and it does not require global clustering of the descriptors found in all images. In addition, EMD can match signatures of different sizes, and it is not very sensitive to the number of clusters—that is, if one cluster is split into several clusters with similar medoids, the magnitude of the EMD is not greatly affected. This is a very important property, since the automatic selection of the number of clusters remains a largely unsolved problem. Finally, recall that the proposed texture representation is designed to work with multiple channels corresponding to different affine-invariant region detectors (here, the H and L operators). Each channel generates its own signature representation for each image in the database, and therefore its own EMD value for any pair of images. We have experimented with several methods of combining the EMD matrices of the separate channels to arrive at a final estimate of the distance between each pair of images. Empirically, simply adding the distances produces the best results.



T0 T1 T2 T3 T4 T5 T6 T7 T8 T9



(a) Spin images



(b) Gabor-like filters

Class	H	L	H+L
T0	0.98	0.89	0.98
T1	0.76	0.71	0.83
T2	0.87	0.90	0.95
T3	1.00	0.93	1.00
T4	0.43	0.52	0.47
T5	1.00	0.93	1.00
T6	0.97	0.84	0.95
T7	0.74	0.71	0.84
T8	0.89	0.63	0.85
T9	0.98	0.91	0.99
Mean	0.86	0.80	0.89

(c) Classification rates

Fig. 10. Top: Samples of the ten texture classes used in our experiments. Bottom: Retrieval and classification results [14].

We have implemented the proposed approach and conducted experiments with a dataset consisting of 200 images—20 samples each of ten different textured surfaces. Figure 10(top) shows two sample images of each texture. Significant viewpoint changes and scale differences are featured within each class. Several of the classes include additional sources of variability: inhomogeneities in the texture patterns, non-rigid transformations, illumination changes, and unmodeled viewpoint-dependent appearance changes. Figure 10(bottom, left) shows retrieval results using intensity-domain spin images [14] as local image descriptors. Spin images are two-dimensional

histograms encoding the distribution of brightness values. The dimensions are the distance from the center or the origin of the normalized coordinate system of the patch, and the intensity value. Spin images are similar to the SIFT descriptors introduced in the previous section, but avoid estimation of the gradient orientation. Notice that for this dataset, the H channel is more discriminative than the L channel. Adding the EMD estimates provided by the two channels results in improved performance. Figure 10(bottom, center) shows the results obtained using the Gabor-like filters commonly used as image descriptors in texture analysis [34,41] instead of intensity-domain spin images. Figure 10(bottom, right) summarizes the classification results obtained by using five samples from each class as training images. The classification rate for each class provides an indication of the "difficulty" of this class for our representation. The mean classification rate is 89% with two classes achieving 100%, showing the robustness of our system against a large amount of intra-class variability. Performance is very good for the rather inhomogeneous textures T5 and T6, but class T4 is not recognized very well, which is probably explained by the lack of an explicit model for viewpoint-dependent appearance changes caused by non-Lambertian reflectance and fine-scale 3D structure.

3.2. *Generative models*

The previous section demonstrated the adequacy of our image descriptors in simple texture classification tasks. Here we go further and introduce generative models for the distribution of these descriptors, along with co-occurrence statistics for nearby patches. At recognition time, initial probabilities computed from the generative model are refined using a relaxation step that incorporates co-occurrence statistics learned at modeling time.

In the supervised framework, the training data consists of single-texture sample images from classes with labels C_ℓ. The class-conditional densities $p(\boldsymbol{x}|C_\ell)$ can be estimated using all the feature vectors extracted from the images belonging to class C_ℓ. We model class-conditional densities as $p(\boldsymbol{x}|C_\ell) = \sum_{m=1}^{M} p(\boldsymbol{x}|c_{\ell m})\, p(c_{\ell m})$, where the components $c_{\ell m}$ are thought of as *sub-classes* and each $p(\boldsymbol{x}|c_{\ell m})$ is assumed to be a Gaussian. The EM algorithm is used to estimate the parameters of the mixture model and is initialized with the output of the K-means algorithm. We limit the number of free parameters and control numerical behavior by using spherical Gaussians with covariance matrices of the form $\Sigma_{\ell m} = \sigma_{\ell m}^2 I$.

In situations where it is difficult to obtain large amounts of fully labeled examples, training on incompletely labeled or unlabeled data helps to improve classification performance [26]. The EM framework provides a natural way of incorporating unsegmented multi-texture images into the training set. Suppose we are given a multi-texture image annotated with the set $\mathcal{L}$ of class indices that it contains—that is, each feature vector $\boldsymbol{x}$ extracted from this image has a *label set* of the form $C_{\mathcal{L}} = \{C_\ell | \ell \in \mathcal{L}\}$. To accommodate label sets, we now use all the data simultaneously to estimate a single mixture model with $L \times M$ components. The

estimation process starts by selecting some initial values for model parameters. During the *expectation* or E-step, we use the parameters to compute probabilistic sub-class membership weights given the feature vectors $\boldsymbol{x}$ and the label sets $C_{\mathcal{L}}$: $p(c_{\ell m}|\boldsymbol{x}, C_{\mathcal{L}}) \propto p(\boldsymbol{x}|c_{\ell m})\, p(c_{\ell m}|C_{\mathcal{L}})$, where $p(c_{\ell m}|C_{\mathcal{L}}) = 0$ for all $\ell \notin \mathcal{L}$ and $\sum_{\ell \in \mathcal{L}} \sum_{m=1}^{M} p(c_{\ell m}|C_{\mathcal{L}}) = 1$. During the *maximization* or M-step, we use the computed weights to re-estimate the parameters by maximizing the expected likelihood of the data.

At this stage, each region in the training image is assigned the sub-class label that maximizes the posterior probability $p(c_{\ell m}|\boldsymbol{x}, C_{\mathcal{L}})$. Next, we need to define a neighborhood for a given region which depends on the size and shape of the region. Here we "grow" the ellipse by adding a constant absolute amount (15 pixels in the implementation) to the major and minor axes. We can now effectively turn the image into a directed graph with arcs emanating from the center of each region to other centers that fall within its neighborhood. The existence of an arc from a region with sub-class label c to another region with label c' is a joint event (c, c') (note that the order is important since the neighborhood relation is not symmetric). For each possible pair of labels, we estimate $p(c, c')$ from the relative frequency of its occurrence, and also find the marginal probabilities $\hat{p}(c) = \sum_{c'} p(c, c')$ and $\check{p}(c') = \sum_{c} p(c, c')$. Finally, we compute the values

$$r(c, c') = \frac{p(c, c') - \hat{p}(c)\,\check{p}(c')}{\left[\left(\hat{p}(c) - \hat{p}^2(c)\right)\left(\check{p}(c') - \check{p}^2(c')\right)\right]^{\frac{1}{2}}}$$

representing the correlations between the events that the labels c and c', respectively, belong to the source and destination nodes of the same arc. The values of $r(c, c')$ must lie between -1 and 1; negative values indicate that c and c' rarely co-occur as labels at endpoints of the same edge, while positive values indicate that they co-occur often. In our experiments, we have found that the values of $r(c, c')$ are reliable only in cases when c and c' are sub-class labels of the same class C. We set $r(c, c')$ to a constant negative value that serves as a "smoothness constraint" in the relaxation algorithm described next, whenever c and c' belong to different classes.

We have implemented the probability-based iterative relaxation algorithm described in the classic paper by Rosenfeld et al. [28] to enforce spatial consistency. The initial estimate of the probability that the ith region has label c, denoted $p_i^{(0)}(c)$, is obtained from the learned Gaussian mixture model as the posterior probability $p(c|\boldsymbol{x}_i)$. Note that since we run relaxation on unlabeled test data, these probabilities must be computed for all $L \times M$ sub-class labels corresponding to all possible classes. At each iteration, new probability estimates $p_i^{(t+1)}(c)$ are obtained by updating the current values $p_i^{(t)}(c)$ using the equation

$$p_i^{(t+1)}(c) = \frac{p_i^{(t)}(c)\left[1 + q_i^{(t)}(c)\right]}{\sum_c p_i^{(t)}(c)\left[1 + q_i^{(t)}(c)\right]}, \quad q_i^{(t)}(c) = \sum_j w_{ij} \left[\sum_{c'} r(c, c')\, p_j^{(t)}(c')\right]$$

The scalars w_{ij} are weights that indicate how much influence region j exerts on

region i. We treat w_{ij} as a binary indicator variable that is nonzero if and only if the jth region belongs to the ith neighborhood. Note that the weights are required to be normalized so that $\sum_j w_{ij} = 1$.

Excellent results for classification/segmentation are obtained for images of an indoor scene and pictures of wild animals. Our first data set contains seven different textures present in a single indoor scene, see Figure 11. Class labels are assigned automatically to each region. Regions of a class are shown in the corresponding column. The second and third rows of Figure 11 show the improvement between the initial labeling and final labeling after relaxation which takes into account the spatial layout. Our second data set consists of unsegmented images of three kinds of animals: cheetahs, giraffes, and zebras. The training set contains 10 images from each class. To account for the lack of segmentation, we introduce an additional "background" class, and each training image is labeled as containing the appropriate animal and the background. Typical classification examples are shown in Figure 12.

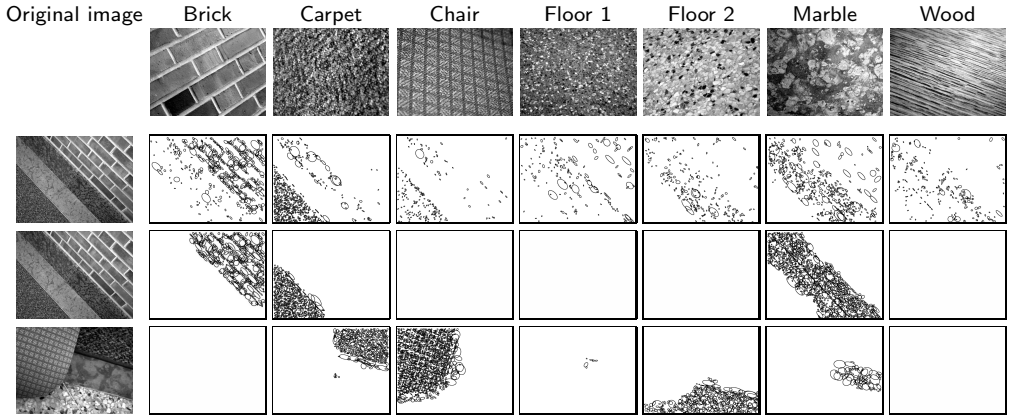


Fig. 11. Segmentation/classification results. From top to bottom: a sample image of each texture class from an indoor scene; initial labeling for an indoor image vs. the final labeling after relaxation– note the significant improvement–; another successful indoor image segmentation (final labeling). See [13] for additional results.

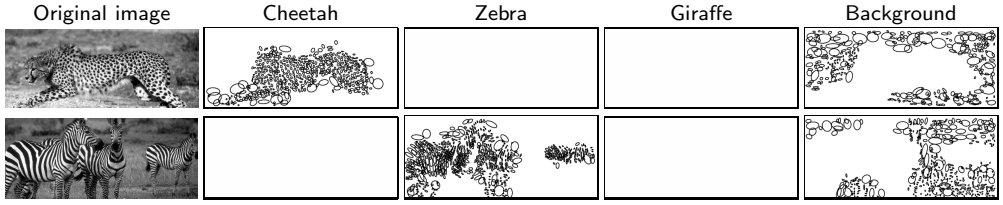


Fig. 12. Segmentation/classification results (final results after relaxation). Animal image classification examples.

4. Recognizing object classes

Learning and recognizing object class models from images is one of the most difficult problems in computer vision. The combination of image description and machine learning/pattern classification techniques has recently led to significant progress. Several recent approaches to category-level object recognition use classification techniques to acquire part models from training images, and then train a classifier to recognize objects. This paradigm has been successfully applied to the recognition of cars [1], faces [36] and human beings [25] in complex imagery. However, the image descriptors used in these methods enjoy very limited invariance properties (mostly translational invariance), which severely limits the range of admissible viewpoints that they can handle. This can be avoided by using local invariants as image descriptors [7,8,15,27]. The approach in section 4.1 constructs parts from individual local invariants, whereas the one presented in section 4.2 uses sets of local invariant features described by their appearance and neighbourhood relations to learn parts.

4.1. *Discriminative local parts*

In this section we demonstrate the power of using invariant local features for building discriminative local parts. The idea is to find groups (clusters) of similar local features, i.e. local parts, and to select among these parts the ones which best discriminate between positive and negative images. Figure 13 shows two discriminative parts for the categories airplane, motorbike, wild cat and person.

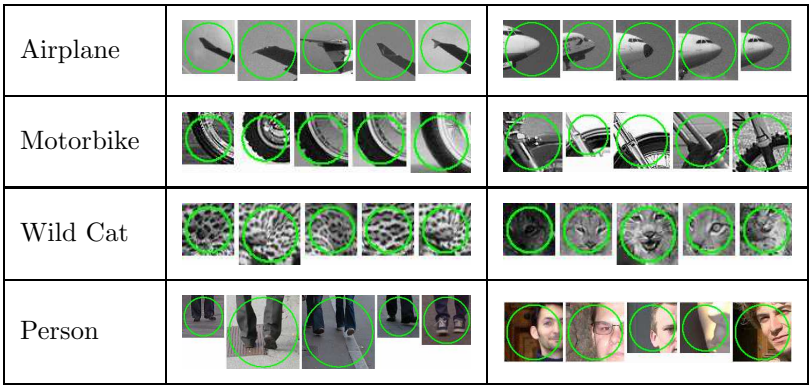


Fig. 13. Examples of discriminative parts (clusters) for the categories airplane, motorbike, wild cat and person. Parts are chosen from the 10 most discriminative ones. Each cluster is represented by five of its members. The circles represent the regions extracted by the scale-invariant detector.

The approach is weakly supervised, that is the training images are labeled as positive and negative, but the objects are not labeled in the positive images. The training set is split in a "clustering" set and a "validation" set. Parts are learned based on the "clustering" set and the significance of each part is determined with

the "validation" set.

The first step of our approach is to extract local invariant features. Here we use Harris-Laplace and Harris-Affine as well as the entropy detector by Kadir and Brady [11] and describe the regions by the SIFT descriptor. The descriptors of the "clustering" set are then used to learn the individual parts. We estimate the Gaussian mixture model of their distribution and each component of the mixture represents a part (cluster). The EM algorithm is used to estimate the parameters of the mixture model and is initialized with the output of the K-means algorithm. Descriptors are assigned to the components with the maximum posteriori probability. We then select parts with the "validation" set. We first compute probabilities $p(C_i|\boldsymbol{x})$ for each descriptor of this set, i.e. determine the probability for each part C_i (component of the Gaussian mixture model). Each part C_i is then ranked by the likelihood ratio between the descriptors of the positive images $\{\boldsymbol{x}_j^u\}$ –note that the individual descriptors are unlabeled– and the negative images $\{\boldsymbol{x}_k^n\}$: $R = \sum_j p(C_i|\boldsymbol{x}_j^u)/\sum_k p(C_i|\boldsymbol{x}_k^n)$. Other criteria can be used, such as mutual information [7]. The final classifier then sets the n highest ranked parts as positive and the others as negative. A descriptor is classified as an object descriptor if it is labeled as belonging to one of the top n parts (maximum posteriori probability for that class) where n is a parameter of our approach.

Our approach is evaluated in two different ways. We first verify that the positive descriptors lie mostly on the object. Figure 14 shows the results for a few test images. Note that the results are very good. Only a few points are incorrectly classified and they could easily be eliminated by any simple coherence criterion. We have also verified that (as in Section 2.3) SIFT features again give significantly better results than steerable filters.

We then evaluate the performance by image classification, that is if the image contains the object or not. This is a standard criterion which allows comparison with existing methods. We report and compare image classification results in table 1. Training and test images are the same as in Fergus et al. [8] and Opelt et al. [27]. We measure performance with the Receiver Operating Characteristic (ROC) equal error rate. It is defined to be the point on the ROC curve–obtained by varying the number of parts n– where the proportion of true positives is equal to the proportion of true negatives. Classification requires an additional parameter, namely the minimum number p of positive descriptors in the image for which the image is classified as positive. The "best" p is estimated from the validation set. Note that this value doesn't necessary lead to the best classification performance on the test set. Table 1 shows that our model in combination with the two scale-invariant detectors Harris-Laplace and the entropy detector outperforms the other methods.

4.2. *Affine-invariant semi-local parts*

In this section we use characteristic patterns formed by affine-invariant patches and semi-local spatial relations to describe salient object parts. Figure 15 illustrates this

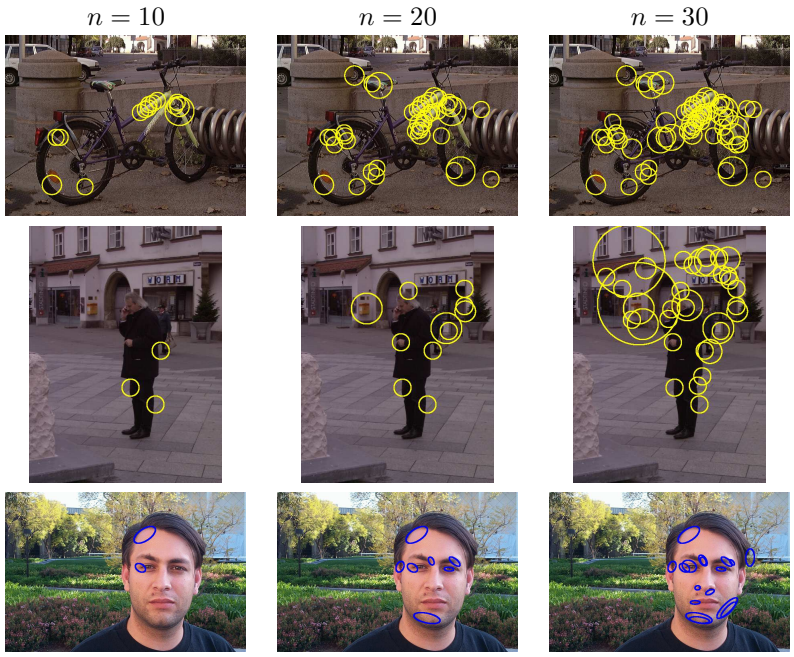


Fig. 14. Positive detections with increasing n for different categories and detectors. First and second row: entropy detector. Third row: Harris-Affine detector.

	Our model	Fergus *et al.* [8]	Opelt *et al.* [27]
airplanes	0.985	0.902	0.889
faces	0.991	0.964	0.935
motorbikes	0.995	0.925	0.922
wildcats	0.87	0.900	—
bikes	0.88	—	0.865
people	0.88	—	0.808

Tab. 1. Image classification performance measured with the equal error rate.

idea with an affine-invariant semi-local part found between face images using the output of the affine-invariant Laplacian and a variant of affine alignment [3]. Note that the part is stable despite large viewpoint variations and appearance changes.

Affine-invariant semi-local parts are geometrically stable configurations of multiple affine-invariant regions, found by the affine Laplace detector. These parts are approximately affinely rigid by construction, i.e. the mapping between instances of the same part in two images can be well represented by a 2D affine transformation. Combined with the *locality* of the parts, this property makes our method suitable for modeling a wide range of 3D transformations, including viewpoint changes and non-rigid deformations. Furthermore, they are more distinctive than individual features

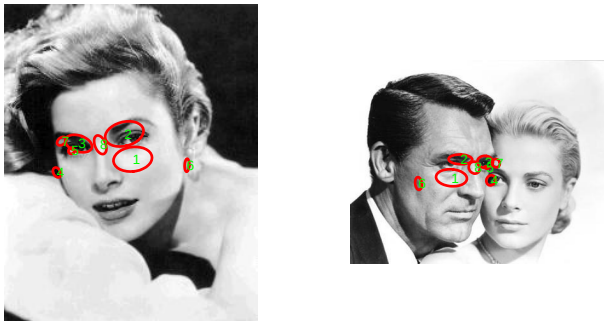


Fig. 15. Matching faces: an affine-invariant semi-local part.

used in the previous section. The mechanism for learning affine-invariant semi-local parts is based on the idea that a direct search for visual correspondence is key to successful recognition. Thus, at training time we seek to identify groups of neighboring affine regions whose appearance and spatial configuration remains stable across multiple instances. To avoid the prohibitive complexity of establishing simultaneous correspondence across the whole training set, we separate the problem into two stages: Parts are initialized by matching pairs of images and then matched against a larger *validation set*. Even though finding optimal correspondence between features in two images is still intractable, effective sub-optimal solutions can be found using non-exhaustive constrained search.

Fig. 16. The butterfly dataset. Two samples of each class are shown in each column.

We have implemented the approach [15] and conducted experiments for the challenging application to the automated acquisition and recognition of butterfly models in heavily cluttered natural images. Figure 16 shows two samples for the seven classes in our dataset which is composed of 619 butterfly images. Note that we don't use any negative images for modeling. The pictures, which are collected from the Internet, are extremely diverse in terms of size and quality. Motion blur, lack of focus, resampling and compression artifacts are common. This dataset is appropriate for exercising the descriptive power of semi-local affine parts, since the geometry of a butterfly is locally planar for each wing (though not globally planar). In addition, the species identity of a butterfly is determined by a basically stable geometric wing

pattern, though appearance can be significantly affected by variations between individuals, lighting, and imaging conditions. It is crucial to point out that butterfly recognition is beyond the capabilities of many current state-of-the-art recognition systems [1,8].

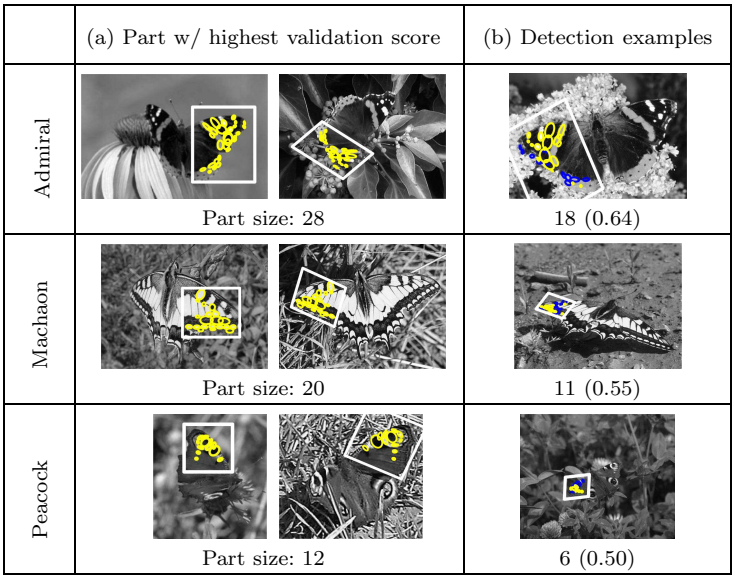


Fig. 17. Butterfly modeling and detection examples. (a) The part with the highest validation score for a class. The part size is listed below each modeling pair. (b) Example of detecting the part from (a) in a single test image. Detected regions are shown in yellow and occluded ones are reprojected from the model in blue. The total number of detected regions (absolute repeatability) and the corresponding repeatability ratio are shown below each image.

The recognition framework is straightforward. Matching and validation are used to identify a fixed-size collection of parts for representing the classes. Candidate parts are formed by matching between eight randomly chosen pairs of training images. Ten verification images per class are used to rank candidate parts according to their repeatability score, and the top ten parts per class are retained for recognition. Figure 17(a) shows the part having the highest repeatability for each of the classes. At testing time, the parts for all classes are detected in each training image. Though multiple instances of the same part may be found, we retain only the single instance with highest number of detected regions. Figure 17 (b) shows examples of part detections in individual test images. The score for a class is defined as the cumulative *relative repeatability* of all its parts, or the total number of regions detected in all parts of the classes divided by the sum of part sizes. For multi-class classification, each image is assigned to the class having the maximum score. Figure 18(a) shows classification results obtained using the above approach (the average rate is 90.4%), and Figure 18(b) shows how performance is improved by using multiple parts.

(a)

Class	Part size	Test images	Correct (rate)
Admiral	179 (12/28)	85	74 (0.871)
Swallowtail	252 (18/29)	16	12 (0.750)
Machaon	148 (12/21)	57	55 (0.965)
Monarch 1	289 (14/67)	48	35 (0.729)
Monarch 2	275 (19/36)	58	53 (0.914)
Peacock	102 (8/14)	108	108 (1.000)
Zebra	209 (16/31)	65	58 (0.892)
Total		437	395 (0.904)

(b)

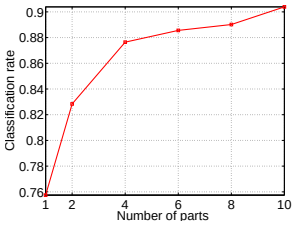


Fig. 18. Classification results for the butterflies. (a) The second column shows the size of the model (top 10 parts) for each class (the sum of sizes of individual parts), and the size of the smallest and the largest parts are listed in parentheses. (b) Classification rate vs. number of parts.

5. Conclusion and discussion

In this chapter we have presented a state of the art on local invariant photometric regions and descriptors. We have shown that the resulting local photometric invariants are very well adapted to object recognition. Research on invariant regions and their description is now well advanced and these invariant features are building blocks for general recognition systems. Of course, improvements of detectors and descriptors are still possible, for example by developing different types of detectors and different region-based descriptors.

The remaining open problems are the recognition of a large number of objects and the recognition of object categories. We think that both problems require the use of machine learning/pattern classification techniques. To handle a large number of objects we have to structure the data based on data reduction, clustering and data mining techniques. Recognizing object categories requires the description of intra-class variation, feature selection, a flexible description of the spatial structure as well as a hierarchical organization of the categories.

Acknowledgments

This research was supported by the European Projects VIBES and LAVA, by a UIUC-CNRS collaboration agreement, by the UIUC Campus Research Board and by the National Science Foundation grants IRI-990709 and IIS-0308087. We are also grateful to D. Lowe, T. Kadir, F. Schaffalitzky and A. Zisserman for providing the code for their detectors and descriptors. Bike and people images have been provided by A. Opelt [27], the airplane, face, motorbike and wild cat images by R. Fergus [8] and the Valbonne images by the INRIA group RobotVis.

References

1. S. Agarwal and D. Roth. Learning a sparse representation for object detection. In *European Conference on Computer Vision*, vol. IV, pp. 113–127, 2002.
2. L. Alvarez and F. Morales. Affine morphological multiscale analysis of corners and multiple junctions. *International Journal of Computer Vision*, 2(25):95–107, 1997.
3. R. Basri and S. Ullman. The alignment of objects with smooth surfaces. In *International Conference on Computer Vision*, pp. 482–488, 1988.

4. A. Baumberg. Reliable feature matching across widely separated views. In *Computer Vision and Pattern Recognition*, vol. I, pp. 774–781, 2000.
5. P. Brodatz. *Textures: A Photographic Album for Artists and Designers*. Dover Publications, New York, 1966.
6. J.L. Crowley and A.C. Parker. A representation for shape based on peaks and ridges in the difference of low pass transform. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 6(2):156–170, 1984.
7. G. Dorko and C. Schmid. Selection of scale-invariant parts for object class recognition. In *International Conference on Computer Vision*, vol. 1, pp. 634–640, 2003.
8. R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Computer Vision and Pattern Recognition*, vol. II, pp. 264–271, 2003.
9. L.M.T. Florack, B. ter Haar Romeny, J.J Koenderink, and M.A. Viergever. General intensity transformation and differential invariants. *Journal of Mathematical Imaging and Vision*, 4(2):171–187, 1994.
10. W.T. Freeman and E.H. Adelson. The design and use of steerable filters. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 13(9):891–906, 1991.
11. T. Kadir and M. Brady. Scale, saliency and image description. *International Journal of Computer Vision*, 45(2):83–105, 2001.
12. K. Kanatani. Geometric information criterion for model selection. *International Journal of Computer Vision*, 26(3):171–189, 1998.
13. S. Lazebnik, C. Schmid, and J. Ponce. Affine-invariant local descriptors and neighborhood statistics for texture recognition. In *International Conference on Computer Vision*, vol. 1, pp. 649–655, 2003.
14. S. Lazebnik, C. Schmid, and J. Ponce. Sparse texture representation using affine-invariant neighborhoods. In *Computer Vision and Pattern Recognition*, vol. 2, pp. 319–324, 2003.
15. S. Lazebnik, C. Schmid, and J. Ponce. Semi-local affine parts for object recognition. In *British Machine Vision Conference*, 2004.
16. T. Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):79–116, 1998.
17. T. Lindeberg and J. Gårding. Shape-adapted smoothing in estimation of 3D shape cues from affine deformations of local 2D brightness structure. *Image and Vision Computing*, 15(6):415–434, 1997.
18. F. Liu and W. Picard. Periodicity, directionality, and randomness: Wold features for image modeling and retrieval. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 18(7):722–733, 1996.
19. D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
20. J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *British Machine Vision Conference*, pp. 384–393, 2002.
21. K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In *International Conference on Computer Vision*, vol. 1, pp. 525–531, 2001.
22. K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In *European Conference on Computer Vision*, vol. I, pp. 128–142, 2002.
23. K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. In *Computer Vision and Pattern Recognition*, vol. 2, pp. 257–263, 2003.
24. K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.

25. K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *European Conference on Computer Vision*, vol. I, pp. 69–81, 2004.
26. K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2–3):103–134, 2000.
27. A. Opelt, M. Fussenegger, A. Pinz, and P. Auer. Weak hypotheses and boosting for generic object detection and recognition. In *European Conference on Computer Vision*, vol. II, pp. 71–84, 2004.
28. A. Rosenfeld, R.A. Hummel, and S.W. Zucker. Scene labeling by relaxation operations. IEEE *Transactions on Systems, Man and Cybernetics*, 6:420–433, 1976.
29. F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce. 3D object modeling and recognition using affine-invariant patches and multi-view spatial constraints. In *Computer Vision and Pattern Recognition*, vol. 2, pp. 272–277, 2003.
30. C. Rothwell, A. Zisserman, D. Forsyth, and J. Mundy. Canonical frames for planar object recognition. In *European Conference on Computer Vision*, pp. 757–772, 1992.
31. Y. Rubner, C. Tomasi, and L.J. Guibas. A metric for distributions with applications to image databases. In *International Conference on Computer Vision*, pp. 59–66, 1998.
32. F. Schaffalitzky and A. Zisserman. Viewpoint invariant texture matching and wide baseline stereo. In *International Conference on Computer Vision*, vol. II, pp. 636–643, 2001.
33. F. Schaffalitzky and A. Zisserman. Multi-view matching for unordered image sets. In *European Conference on Computer Vision*, vol. I, pp. 414–431, 2002.
34. C. Schmid. Weakly supervised learning of visual models and its application to content-based retrieval. *International Journal of Computer Vision*, 56(1):7–16, 2003.
35. C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–534, 1997.
36. H. Schneiderman and T. Kanade. A statistical method for 3D object detection applied to faces and cars. In *Computer Vision and Pattern Recognition*, vol. I, pp. 746–751, 2000.
37. M.J. Swain and D.H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.
38. M. Turk and A. Pentland. Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition*, pp. 586–591, 1991.
39. T. Tuytelaars and L. Van Gool. Matching widely separated views based on affine invariant regions. *International Journal of Computer Vision*, 59(1):61–85, 2004.
40. L. Van Gool, T. Moons, and D. Ungureanu. Affine / photometric invariants for planar intensity patterns. In *European Conference on Computer Vision*, pp. 642–651, 1996.
41. M. Varma and A. Zisserman. Classifying images of materials: Achieving viewpoint and illumination indepence. In *European Conference on Computer Vision*, vol. III, pp. 255–271, 2002.
42. K. Xu, B. Georgescu, D. Comaniciu, and P. Meer. Performance analysis in content-based retrieval with textures. In *International Conference on Pattern Recognition*, vol. IV, pp. 275–278, 2000.