Object Class Recognition Using Discriminative Local Features

Gyuri Dorkó, and Cordelia Schmid, *Senior Member, IEEE,*

Abstract

In this paper, we introduce a scale-invariant feature selection method that learns to recognize and detect object classes from images of natural scenes. The first step of our method consists of clustering local scale-invariant descriptors to characterize object class appearance. Next, we train *part classifiers* on the groups, and perform feature selection to determine the most discriminative parts. We use local regions to realize robust and sparse part and texture selection invariant to changes in scale, orientation and affine deformation and, as a result, we avoid image normalization in both training and prediction phases. We train our object models without requiring image parts to be labeled or objects to be separated from the background. Moreover, our method continues to work well when images have cluttered background and occluded objects. We evaluate our method on seven recently proposed datasets, and quantitatively compare the effect of different types of local regions and feature selection criteria on object recognition. Our experiments show that local invariant descriptors are an appropriate representation for many different object classes. Our results also confirm the importance of appearance-based discriminative feature selection.

Index Terms

object recognition, feature evaluation and selection

I. Introduction

Recognizing classes of objects is one of the fundamental challenges in computer vision. Recently proposed techniques in vision and machine learning have led to significant improvements [1]–[4], however many of these methods are limited to fixed size windows or require hand-segmented, pre-normalized training and test images [5]–[7]. In this paper, we exploit state-of-the-art learning techniques and recent advances in computer vision to

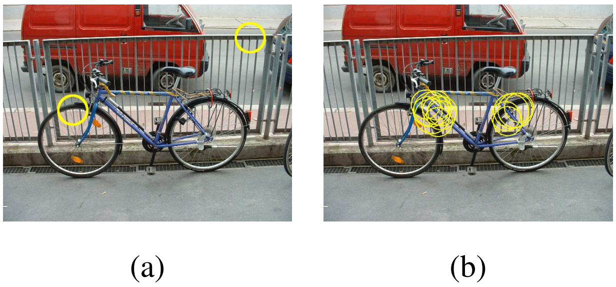


Fig. 1. Illustration of feature selection. (a) Two similar regions which cannot be used in a purely appearance based system to distinguish between the bicycle and the background. (b) The most discriminative features of the bicycle determined by our method.

develop discriminative feature selection for object-part recognition and detection. Our two-step approach extracts scale- and affine-invariant local features from unnormalized images and trains a generative class model using these. The approach is "weakly supervised" in the sense that images with positive examples are labeled but the objects in them are not marked or segmented, and are present in arbitrary non-registered locations in cluttered scenes. Moreover, each positive training image can contain multiple instances of the same object class with a large heterogeneous background. Our method is invariant to viewpoint changes, without requiring alignment or pre-normalization of images.

The bicycle example in Fig. 1 illustrates the importance of discriminative feature selection. In (a), the two regions denoted by circles are selected from the output of a scale-invariant operator [8] for illustration purposes. Even though one of them lies on the background and the other on the object ("bicycle"), by inspection and likewise in the description space they are very similar. It turns out that this region is not discriminative for the bicycle class — it occurs regularly with small tubular or transparent parts, and with "donut-like" patches. Fig. 1(b) shows the final result of our feature selection method; the circles correspond to the most discriminative regions selected from the output of the operator [8].

We now outline our approach. The training set contains images labeled as positive and negative. We mark an image as positive if at least one instance of the object class is found in the image. Negative images contain only background. The first step consists of extracting local scale-invariant features from the training images. As the positive images also contain background clutter, the extracted features can belong to either objects or background, and are thus unlabeled. To produce a model we cluster the features and we construct initial probabilistic *part classifiers* from the resulting groups (Section III-A), then refine these using various ranking methods (Section III-B). The n highest ranking classifiers are selected, and used for detection of discriminative parts in unseen images. Ranking requires an unseen validation set of descriptors provided by extracting features from the remaining portion of the training set.

In previous work [9], we applied the ranking methods described in this paper to strictly supervised environments. This paper extends our approach to scenarios with weak supervision, and validates it with extensive experiments on commonly used databases. Importantly, we also demonstrate how to combine different detectors during selection.

A. Related Work

Most appearance-based approaches to object class recognition characterize the objects by their global appearance, usually the entire image [5], [10]. They are not robust to occlusion and suffer from a lack of invariance to similarity transformations such as scale or rotation. Furthermore, these methods are only applicable to rigid objects and they require either preliminary segmentation or evaluation on multiple windows extracted at different locations and scales. Invariance to changes in viewpoint requires scanning the space of affine transformations, which is computationally very expensive. The high-dimensionality of the representation also limits the application of many standard learning techniques. Local features are an increasingly popular method for overcoming these problems in object detection and recognition.

Weber *et al.* [2] use localized image patches and explicitly compute their joint spatial probability distribution. Recently, Fergus *et. al* [11] extend this approach by learning global models of object classes based on scale-invariant image regions. In this paper, we show that in many applications a purely appearance-based method outperforms [2], [11]. Agarwal and Roth [7] first learn a vocabulary of parts, determine spatial relations on these parts, and use them to train a Sparse Network of Winnows (SNoW) Learning Architecture. Since they learn rigid spatial relations in terms of distance and direction between each pair of parts, their method is invariant neither to scale nor rotation. Leibe and Schiele [3] also learn a codebook of local appearance and relative spatial positions of individual parts, and use a voting scheme to combine them and probabilistically segment unseen images. We also note work on the application of local affine invariant features to related areas, such as texture representation [12] and image retrieval [13].

Some recent methods combine feature selection and local descriptors. Viola and Jones [4] select rectangular Haar-like features using AdaBoost. Chen *et al.* [14] use boosting to construct components by local non-negative matrix factorization. Opelt *et al.* [15] apply Adaboost to learn a local features classifier for determining the presence or absence of objects in images; we compare with their results in Section IV-B. Amit and Geman [16] combine small localized oriented edges with decision trees. Mahamud and Hebert [6] select discriminative object parts and develop an optimal distance measure for nearest neighbor search. Rikert *et al.* [17] use a mixture model that retains only discriminative clusters, and Schmid [18] selects significant texture descriptors in a weakly supervised framework. Both approaches select features based on their likelihood. Ullmann *et al.* [1] use image fragments and combine them with a linear discriminative type classification rule. Their selection algorithm is based on mutual information.

B. Overview

The paper is organized as follows. In Section II we detail our chosen representation and feature extraction method. Section III describes the learning part of our system: estimation of object parts (Section III-A), selection of discriminative parts (Sections III-B and III-C) and construction of the final classifier from discriminative parts (Section III-D). Section IV contains experimental results from seven different databases as well as a discussion of the effect of different parameter settings on our method's performance. Section V summarizes and concludes.

II. LOCAL DESCRIPTORS

Local representations of images are useful to cope with a wide variety of natural scenes containing cluttered background and occluded objects. A local descriptor represents a region or patch of an image. *Interest point detectors* select salient regions (points and their neighborhoods), and with some detectors the results are invariant to scale and/or viewpoint changes. Each selected patch is characterized by a descriptor vector. At this point, one can also impose additional invariances, such as rotation or illumination. In this section, we motivate our choices for local descriptor computation.

A. Detectors

Many different region detectors exist in the literature [8], [19]–[23]. Here we briefly present the ones that we use. Scale invariant detectors select regions at significant locations with a corresponding scale parameter representing the size of the region. The advantage is that features are found at the most informative scales and optionally affine transformations, thereby reducing the complexity of subsequent processes because only a limited number of regions need to be considered. The most important property of such a detector is repeatability, for example, an affine-invariant detector should select nearly the same regions of an object even though it is observed from two different viewpoints.

The Harris-Laplace detector [19] extends the standard Harris operator by applying it to each scale level, then selecting characteristic points in the scale-space using the Laplace operator. An optional affine estimation [20] on the neighborhoods of detected points provides viewpoint invariance. The result of this is stable circular or elliptical regions centered on corner-like structures.

The method of Kadir and Brady [8] extracts blob-like regions (homogeneous or non-homogeneous regions surrounded by edges). It finds circular regions in the image having the highest saliency based on maxima of the entropy scale-space of region histograms. The first row of Fig.13 shows the result of the aforementioned detectors on a sample image.

B. Descriptors

Before transformation to the feature space, we normalize the regions. We interpret the detector output as a location (coordinates of the center) and a neighborhood represented by a circle radius or a parameterized ellipse. We map each point and neighborhood to a general circular region, with appropriate smoothing in the case of down-scaling. We can achieve orientation invariance at this point by rotating the circular patch to the direction of the average gradient measured on a small point neighborhood. Note that this step is indispensable when viewpoint invariance is desired (case of ellipse in detection).

Based on earlier studies [24] and on our own experience we chose the Scale Invariant Feature Transform (SIFT) [22] as a representation of the extracted normalized regions. We retain the standard parameter settings, and compute SIFT on a 4x4 grid with an 8-bin orientation histogram for each cell, resulting in a 128-dimensional real vector for each local region.

III. Learning and Selection

In our approach, object classes are represented as sets of object parts. With each part we associate a classifier learned from similar descriptors. Some classifiers are more reliable than others, because they invoke more discriminative features. Those representative *part classifiers* are chosen by our feature selection method to build a robust and reliable detection system. In this section we describe the learning of simple *part classifiers*, selection of discriminative parts, and construction of a final classifier used as a first step of object class detection, or image classification.

A. Learning Part Classifiers

The first step of the training phase is an unsupervised estimation of a Gaussian mixture model [25] (GMM). The training data is separated into two parts: the *clustering set*, used to estimate the actual GMM, and the *validation set*, used at a later stage for the selection (see Section III-B). The clustering set contains local features extracted from positively labeled training images. Optionally, to ensure sufficient numbers of descriptors, some can be added from our negative images (in which case they are also considered as unlabeled).

We employ a parametric estimation to model the distribution of our local descriptors in our clustering set. Our method is based on a GMM, a linear combination of Gaussian densities $p(\mathbf{x}|C_i)$ expressed as

$$p(\mathbf{x}) = \sum_{i=1}^{K} p(\mathbf{x}|C_i)P(C_i), \tag{1}$$

where K is the number of Gaussian components within the mixture, $P(C_i)$ corresponds to the mixing parameters and $\sum_i^K P(C_i) = 1$. The individual Gaussian components are of the form

$$p(\mathbf{x}|C_i) = \mathcal{N}(\boldsymbol{\mu}_i, |\boldsymbol{\Sigma}_i) \tag{2}$$

where $\boldsymbol{\mu}_i$ is a d dimensional mean vector and $\boldsymbol{\Sigma}_i$ is the $d \times d$ covariance matrix for component C_i. In our case $d = 128$, corresponding to the dimension of the SIFT features.

The model parameters $\boldsymbol{\mu}_i$, $\boldsymbol{\Sigma}_i$ and $P(C_i)$ of (1) and (2) are computed with the *expectation-maximization* (EM) algorithm [25]. EM is initialized with the output of K-means and at each iterative M-step we update the parameters as follows:

$$\boldsymbol{\mu}_i^j = \frac{\sum_{n=1}^{N} P^{j-1}(C_i|\mathbf{x}^n)\mathbf{x}^n}{\sum_{n=1}^{N} P^{j-1}(C_i|\mathbf{x}^n)} \tag{3}$$

$$\boldsymbol{\Sigma}_i^j = \frac{\sum\limits_{n=1}^{N} P^{j-1}(C_i|\mathbf{x}^n)(\mathbf{x}^n - \boldsymbol{\mu}_i^j)(\mathbf{x}^n - \boldsymbol{\mu}_i^j)^T}{\sum_{n=1}^{N} P^{j-1}(C_i|\mathbf{x}^n)} \tag{4}$$

$$P^j(C_i) = \frac{1}{N}\sum_{n=1}^{N} P^{j-1}(C_i|\mathbf{x}^n), \tag{5}$$

where N is the number of unlabeled descriptors ($\mathbf{x}_n$) in the clustering set. We limit the number of free parameters in the optimization by using diagonal covariance matrices, assuming statistical independence of the variables. This restriction also simplifies the computation of (4) and helps to prevent $\mathbf{\Sigma}_i$ from becoming singular. If the model is estimated with k-means, the individual descriptor classification rate drops by an average 4%. The experimental results in the following are given for a model estimated by EM with diagonal covariance matrices.

Fig. 2 shows some selected components of different object classes obtained by assigning the training descriptors to their closest cluster. We show two of the ten best clusters, according to our ranking method (Section III-B). The clusters typically contain representative object parts or textures. For example, for airplanes, the nose has a very characteristic shape as does the tailplane (see Fig. 2, first row). We also got significant clusters on the fuselage containing the small passenger windows, and on the wing. In the case of bicycles and motorbikes, tires, wheels and tubular parts are clearly grouped and distinguished. Faces give one of the most impressive results, as left and right eyes, including the eyebrows, are clustered separately. Sometimes, if objects have very characteristic textures, their corresponding descriptors are clustered together as it is the case for the wild cats in the figure.

Using the mixture model we define a separation boundary for each component to construct *K part classifiers*. Each classifier is associated with an object or background part represented by a single Gaussian. A test feature $\mathbf{y}$ is assigned to the component i^* having the highest probability:

$$i^* = \underset{i}{argmax}\, p(\mathbf{y}|C_i)P(C_i)$$

Fig. 3 shows four examples of separation boundaries based on a GMM with $K = 8$ components. Note that the figure is just an illustration, in practice the number of components are much larger and our feature space is high-dimensional.

B. Selection

The *selection algorithm* ranks the components according to their ability to discriminate between object-class and background. We rank the parts by testing each individual classifier and assign a score according to one of the following two methods.

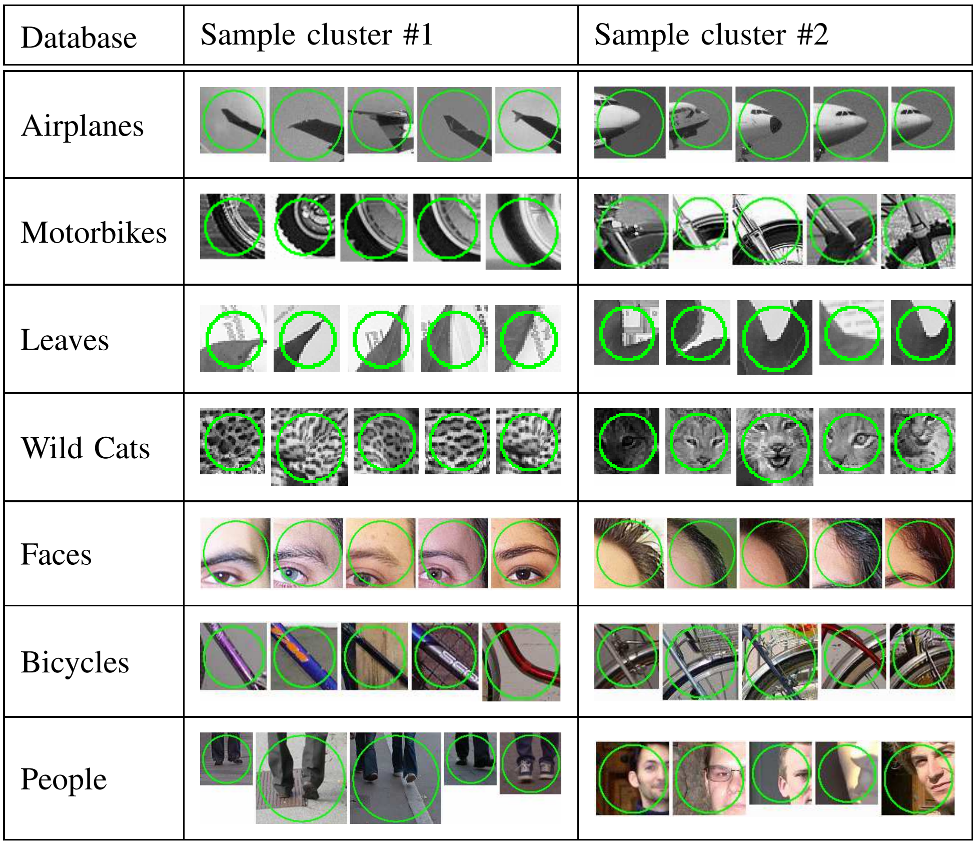


Fig. 2. Two examples of clusters for each object-class used in the experiments. We show 2 of the 10 best clusters for Kadir-Brady interest points.

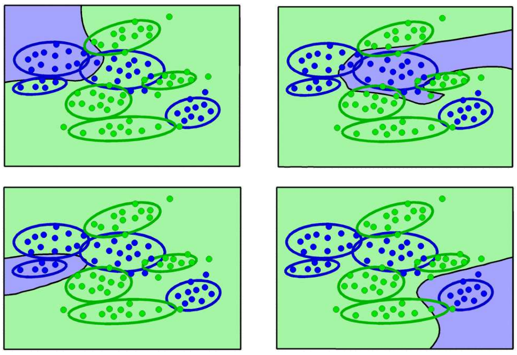


Fig. 3. Illustration of a GMM model with $K = 8$ components. An object part classifier is associated with each component. We only show an illustration of 4 *part classifiers* in 2-dimension. Separation boundaries are shown for each *part classifier*. In practice we have many more *part classifiers* and our features are high-dimensional ($d = 128$).

Independent **ranking by classification likelihood** promotes components having high true positive and low false positive rates. The scores ($R_{\mathcal{L}}(C_i)$) are computed as

$$R_{\mathcal{L}}(C_i) = \frac{\sum_j^{V^{(u)}} P(C_i|\mathbf{v}_j^{(u)})}{\sum_j^{V^{(n)}} P(C_i|\mathbf{v}_j^{(n)})}, \tag{6}$$

where $V^{(u)}$ and $V^{(n)}$ are respectively the numbers of unlabeled (potentially positive) descriptors ($\mathbf{v}_j^{(u)}$) and negative descriptors ($\mathbf{v}_j^{(n)}$) from the *validation set*. Intuitively, this method is well suited for classification and detection purposes because it performs selection by classification rate. This is confirmed by our experiments in Section IV-B. This method is robust to changes in parameter settings and tolerates overfitting in the estimated PDF of the data. On the other hand, $R_{\mathcal{L}}(C_i)$ typically selects very "specific" components i.e. ones near-zero values in the denominator ($P(C|\mathbf{v}_j^{(n)})$). Even though individually these rare parts have low recall rates, combinations of them can provide sufficient recall with excellent precision.

If the main purpose of our system is to produce a sparse object-class representation, it is best to select a few discriminative and *general part classifiers*. Here we use the **mutual information** [26] criterion, which ranks *part classifiers* based on their information content for separating the background from the object-class. The mutual information of component C_i and object-class O is

$$\begin{aligned} R_{\mathcal{I}}(C_i) = {} & P(\bar{C}_i, \bar{O}) \log \frac{P(\bar{C}_i, \bar{O})}{P(\bar{C}_i)P(\bar{O})} \\ & + P(C_i, \bar{O}) \log \frac{P(C_i, \bar{O})}{P(C_i)P(\bar{O})} \\ & + P(\bar{C}_i, O) \log \frac{P(\bar{C}_i, O)}{P(\bar{C}_i)P(O)} \\ & + P(C_i, O) \log \frac{P(C_i, O)}{P(C_i)P(O)} \\ = {} & \sum_{\substack{k=\{C_i,\bar{C}_i\} \\ l=\{O_i,\bar{O}_i\}}} P(k,l) \log \frac{P(k,l)}{P(k)P(l)}. \end{aligned} \tag{7}$$

Note that both C_i and O can be seen as binary events, therefore for simplicity we defined $\bar{C}_i$ and $\bar{O}$ as corresponding negative events. We estimate the probabilities in (7) from the validation set:

$$P(\bar{C}_i, \bar{O}) = \frac{\sum_j^{V^{(n)}} P(\bar{C}_i|\mathbf{v}_j^{(n)})}{V^{(u)} + V^{(n)}} \tag{8}$$

$$P(C_i,\bar{O}) = \frac{\sum_j^{V^{(n)}} P(C_i|\mathbf{v}_j^{(n)})}{V^{(u)} + V^{(n)}} \tag{9}$$

$$P(\bar{C}_i,O) = \frac{\sum_j^{V^{(u)}} P(\bar{C}_i|\mathbf{v}_j^{(u)})}{V^{(u)} + V^{(n)}} \tag{10}$$

$$P(C_i,O) = \frac{\sum_j^{V^{(u)}} P(C_i|\mathbf{v}_j^{(u)})}{V^{(u)} + V^{(n)}} \tag{11}$$

$$P(\bar{C}_i) = P(\bar{C}_i,\bar{O}) + P(\bar{C}_i,O) \tag{12}$$

$$P(C_i) = P(C_i,\bar{O}) + P(C_i,O) \tag{13}$$

$$P(O) = \frac{V^{(u)}}{V^{(u)} + V^{(n)}} \tag{14}$$

$$P(\bar{O}) = \frac{V^{(n)}}{V^{(u)} + V^{(n)}} \tag{15}$$

$$\tag{16}$$

We require

$$P(C_i|O) > P(C_i|\bar{O})$$

so that we select only parts informative for the object-class and not for the background. (14) naively assumes that all unlabeled descriptors in the validation set belong to objects. Owing to similar negatively labeled points, unlabeled background *part classifiers* receive low scores.

C. Combination of Detectors

In Section II we proposed feature selection using two different criteria. Our ranking mechanism offers an elegant way to combine the output of several underlying feature detectors, leading to improved performance. Assuming that the descriptors computed from different detectors are independently distributed, we can separately estimate their GMMs, and construct their *part classifiers* independently. To provide input for the ranking step, we create a validation set for each detector from the same validation images. It is straightforward to adapt equations (6) and (7) for multiple detectors. The normalization factors $V^{(u)}$ and $V^{(n)}$ are the sums of the total number of unlabeled and negative descriptors over all feature types on the validation sets.

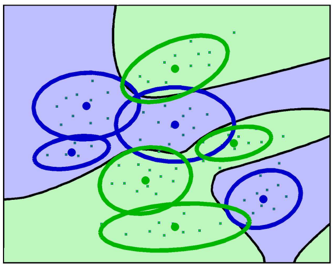


Fig. 4. The final classifier constructed on $K = 8$ GMM model, with $n = 4$ selected *part classifiers*. (See Fig. 3 for the individual *part classifiers*.) The separation boundary indicates if a test feature at that position is classified as positive (object) or as background.

D. Final Feature Classifier

Based on the ranking we learn a final classifier (see Fig. 4). We choose n *part classifiers* of highest rank and mark them as *positive*, where n is the parameter of our system. The rest of the classifiers are negative, firing on negative descriptors and on non-discriminative positive ones. Note that the construction of our *part classifiers* is based on one-to-one relationship between a part and a Gaussian component, thus the MAP criterion only activates one *part classifier* per descriptor.[1]

We now discuss two applications of the final classifier. Our object part detection can act as an initial step for localization within images. For an example see Fig. 13. In the second row we only select the discriminative motorbike parts with our final classifier. The output of the selection is not a binary decision, the ranking step assigns scores to each *part classification* that can be used to determine the certainty of each positive detection.

Another application is classification of the presence or absence of an object class in an image. In this case, besides the number of selected classifier (n) there is an additional criterion to decide whether an instance of the object class can be found on the image. For the experiments below we chose a simple condition with a parameter p to specify the minimum number of detected positive descriptors p, required to label an image as positive. However, other prior or learned knowledge such as neighborhood or geometrical constraints, required scales, etc. can easily be added at this point.

The parameter p has to be carefully chosen. It is set according to the model complexity, the

[1] If this is not the case, as with the SVMs in our earlier work [9], we can classify a descriptor as positive if any (maybe more than one) of the positive *part classifiers* fires on it.

number of selected *part classifiers* (n), the chosen detector and descriptor, and the appearance of the object class. As an example, if the object class contains certain specific parts and they are easily detected, our objects can be built up just from these pieces: A face contains two eyes, a nose, a mouth and some forehead parts. We can expect that each of these few parts are represented by a corresponding *part classifier*, therefore p can be set to a relatively low number without damaging the performance. Conversely, if we have an object class like *Wild Cats*, we can expect the main texture component to be very discriminative. This is confirmed by the sample cluster #1 of *Wild Cats* in Fig. 2 which is the top $R_{\mathcal{L}}$ ranked component of the category. Evidently the texture appears multiple times on the object, therefore the *part classifier* corresponding to such a component selects more than one feature on a test image. In this case, a larger p value gives better performance. In our experiments in Section IV-B we, estimate this parameter p using the validation set.

Naïve Bayes offers another natural way to combine our selected n *part classifiers* to decide whether the object is present in the image. Instead of fixing the minimum number of required detections (p) we have to set a decision boundary on the sum of the log likelihood ratios. As in the case of p, this parameter can be estimated on the verification set. Our experiments showed that the two types of image classification behave very similar, the difference between their average performance was insignificant (0.02%). Since a detailed discussion of the two methods lead to similar conclusions we omit the results of the Naïve Bayes in the section experiments.

IV. Experiments

In this section we present numerical evaluation of our described method in the application areas of "pre-classification for detection" and "image classification" (see Section III-D). For our experiments we used seven different datasets from various sources: airplanes, bicycles, people, motorbikes, leaves, wild cats and faces. All of these sets have been used earlier by others. Example images from the test sets are shown in Fig. 5. To simplify comparisons, we used the same training and test set divisions as others when these were known. With the databases of airplanes, motorbikes and faces our positive training and test images were exactly the same as in [11], however, since our learning method also requires negative training data, we used half of their negative test sets as training while kept the other half as test. For the wild cats database we used the same numbers of training and test images. These were randomly selected from the Corel Image Library *wildcats* category. The leaves database was

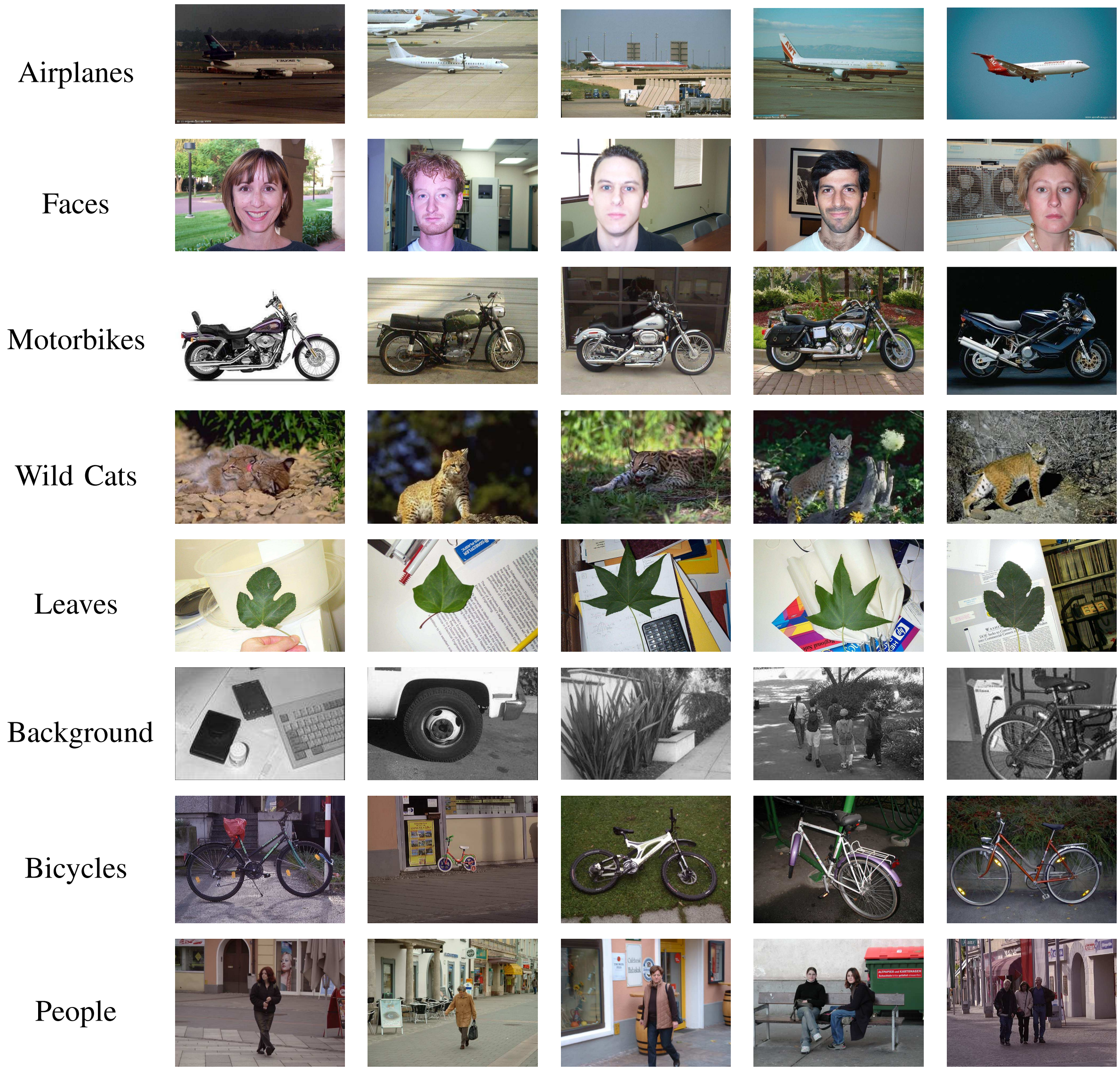


Fig. 5. Samples for test images. Databases Airplanes, Faces, Motorbikes and Leaves with their Background set were obtained from http://www.robots.ox.ac.uk/∼vgg/data. Bicycles and People were downloaded from http://www.emt.tugraz.at/∼pinz/data/. The Wild Cats category is from the Corel Image Library.

used in [27] but a probably with a different negative set. The experiments with bicycles and people had exactly the same positive and negative training and test images as in [15].

As discussed in Section II we chose SIFT [22] descriptors as the representation of our interest regions, but regions were extracted by different detectors. In the reports below we use the following notation. Harris Laplace [19] points are abbreviated as HL, or in case of optional affine invariance [20] HLA. The entropy based detector of Kadir and Brady [8] is denoted by ENTR. To demonstrate the combination of different detectors we combined corner-like (HL) and blob-like regions (ENTR) in our experiments which we refer as COMB.

In all experiments we kept all interest point detector parameters the same, which resulted in between 100 and 300 extracted features per image depending on the database, image sizes, and detectors. The slow speed of the ENTR detector forced us to downscale all images larger than 300 pixels in width or height. We also eliminated regions with very small scales from both ENTR and HL, because we believe that these regions cannot be well represented with high dimensional SIFT descriptors. Unfortunately the background set used from [11] contained images with very few detections which may affect both our results and [11]'s. These experiments are still included because they give a valuable comparison. We tried to keep them as unbiased as possible by keeping the p parameter (the number of required object parts) low. The experiments using other background sets (bicycles and people) are not affected by this as their background sets provided similar numbers of detections to the positive images, i.e. few hundred features per image.

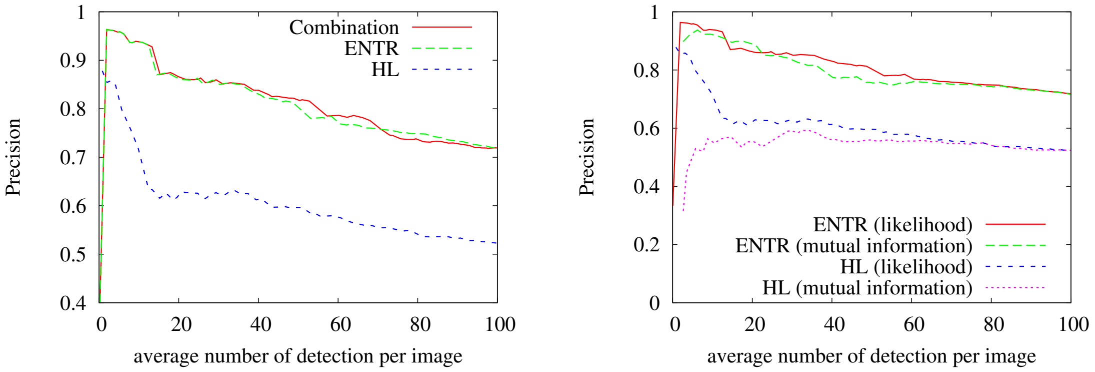
All of the tests were done in a weakly-supervised environment. For each experiment we kept the training set and test sets disjoint, and divided the training set into two disjoint subsets: the clustering set and the validation set. The clustering set was used to estimate GMM as it is discussed in Section III-A. In our bicycle experiments we add features from the negative training images to the clustering set, while in other cases we used the negative features only for validation.

A. *Pre-classification for detection*

The following experiments measure the performance of our final classifier on marked test images. For these experiments we used the *bicycles* database. As described earlier the classifier was trained in a weakly supervised fashion. To produce the ground truth we hand segmented all of the bicycles in the test images and marked a selected feature as true positive if its center was located on the object.

These experiments allowed us to verify that our model did indeed learn the object class correctly, and also to compare the results of different methods. Notice that we expect only a subset of the descriptors on the object to be classified as positive, and that such low *recall*[2] rates do not always imply poor performance – they may just indicate the presence of large numbers of non-discriminative features on the objects. The most important factor for us is

[2] Recall is the ratio of true positives to true features (true positives + true negatives).



(a) (b)

Fig. 6. The precision of the detected features on the bicycle database. (a) evaluates the two detectors and their combination with the ranking method $R_{\mathcal{L}}$. (b) compares the two different ranking methods for the individual detectors.

the *precision*:

$$\text{Precision} = \frac{\text{True positives}}{\text{True positives} + \text{False positives}},$$

which clearly indicates that how many of our selected features are actually object descriptors. Naturally, we also favor more detections with the same precision, but once again, the recall-rate is not a very suitable measure for this. Results achieving both high precision and high recall with only one or two detections in total are not considered to achieve good performance. Therefore instead of an RPC[3] curve we show the *precision* as a function of the average number of detections per image. This provides a realistic comparison of the different interest point detectors in a scale-invariant environment.

Fig. 6 (a) shows the classification results for the two individual detectors on unseen test images from the bicycles database. The ENTR detector was the most precise in this case. E.g. an average selection of 30 points per image provided 85% precision with ENTR, but only 62% with HL. The combination of the two detectors (COMB) produced similar performance to ENTR alone, because the significant performance difference in the individual results caused the combined ranking to choose mainly ENTR *part classifiers*.

To compare the performance of different ranking methods, we show a similar figure (Fig. 6 (b)) on the same dataset using the two individual detectors with the two different ranking methods. For ENTR, the mutual information performed the same, sometimes slightly

[3] Recall Precision Curves are one of the common measures for object detection and database retrieval. Here we have to use a different measure because recall relies on the number of true descriptors, and these are different for each detector.

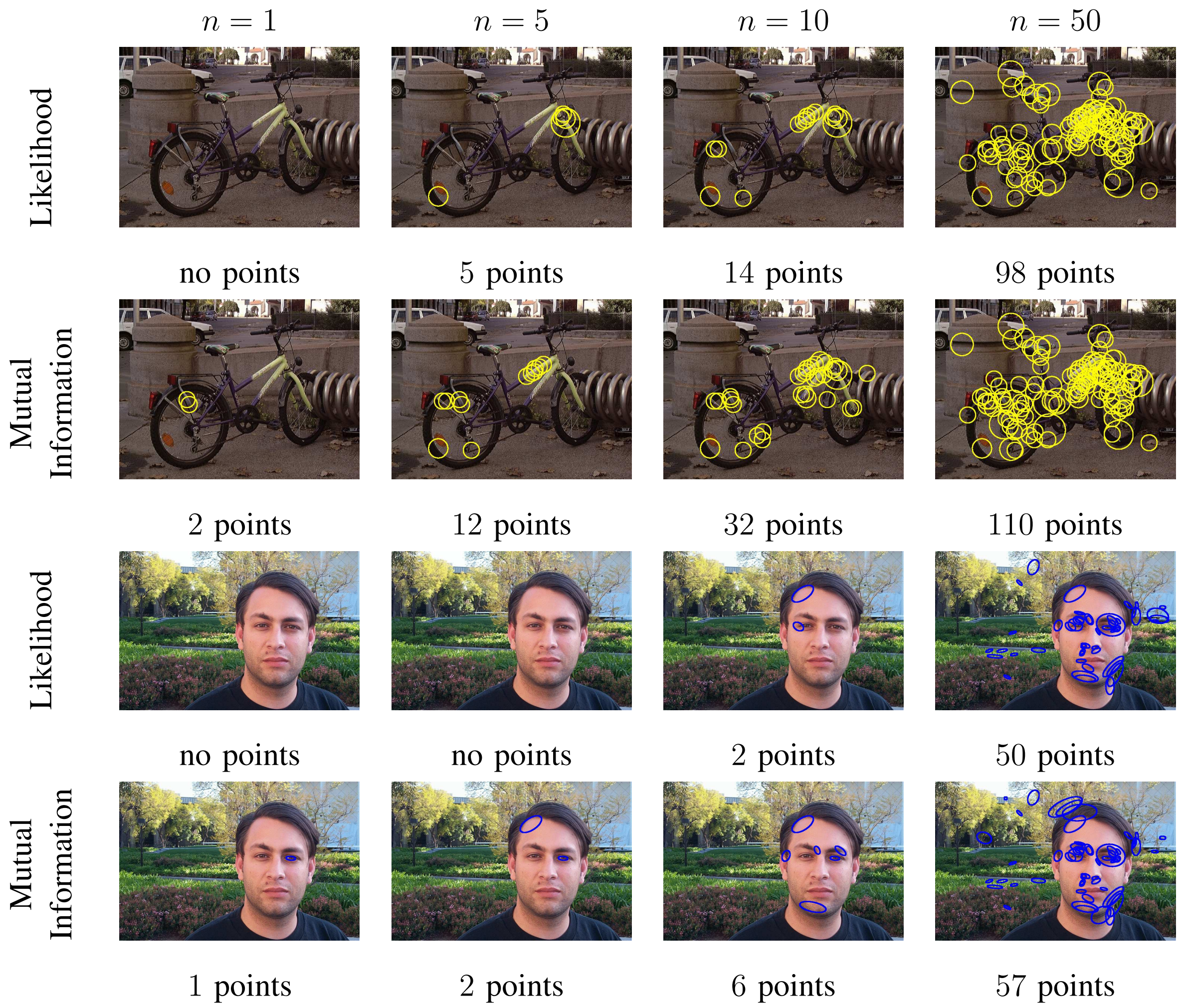


Fig. 7. Examples of feature selection with increasing n. Mutual information selects more informative clusters in the first place, which leads to more positive detections with small n.

worse or slightly better. But for HL the mutual information was always below the likelihood, because $R_{\mathcal{L}}$ selected some specific (very precise) *part classifiers* in the first place, which led to relatively high performance.

Even though mutual information did not perform as well as likelihood, overall it has some important benefits that are illustrated by the examples in Fig. 7. Notice, that in the first ($n = 1$) column of the figure, our final classifier has already selected some features on the test image using only the top ranked cluster. As a general rule the top n *part classifiers* mark more regions with mutual information than with likelihood. $R_{\mathcal{L}}$ ranking prefers accurate but specific[4] *part classifiers* to general ones. Whereas $R_{\mathcal{I}}$ selects more "general" informative clusters, and is thus preferable in applications that require *focus of attention* mechanisms or sparse representations of the feature space. Besides the bicycles and faces, similar examples can be found for the people in Fig. 8. We also noticed that in the case of people (Fig. 8) there was no large difference between the top n *part classifiers* in performance i.e there are no very specific or very general features.

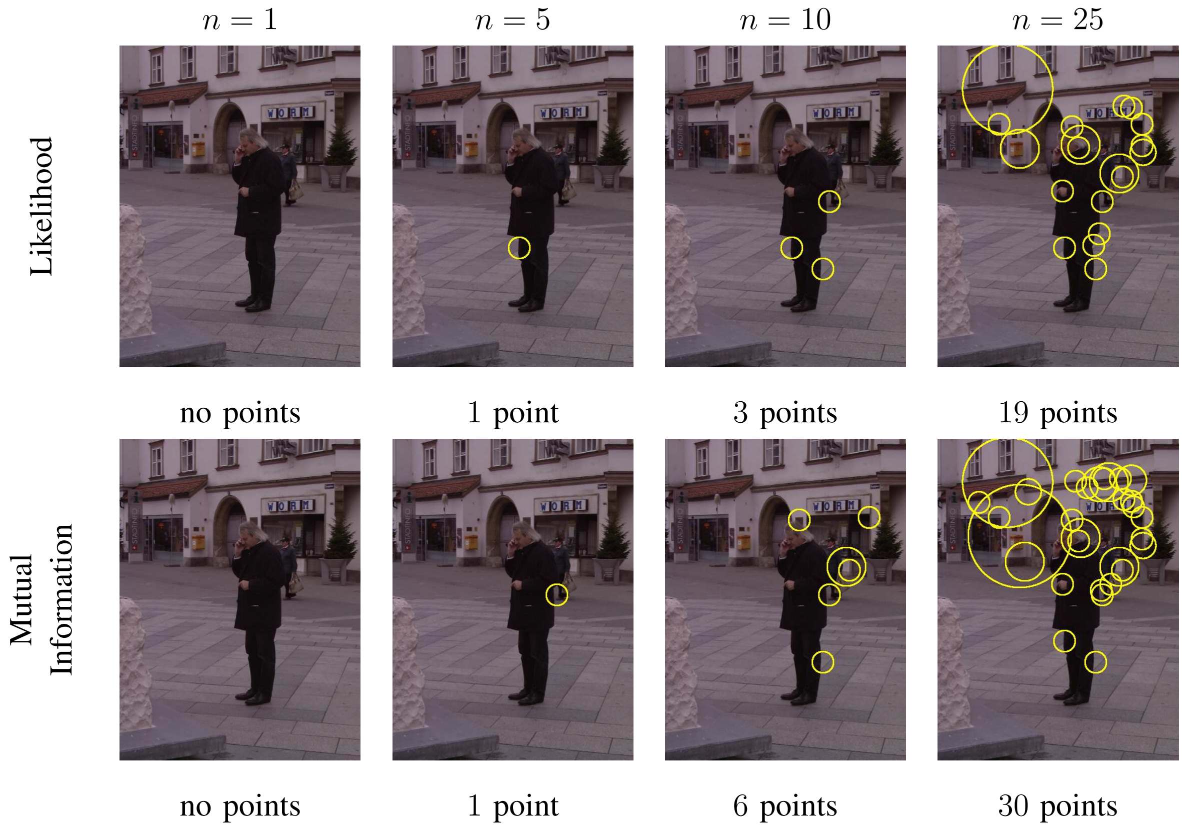
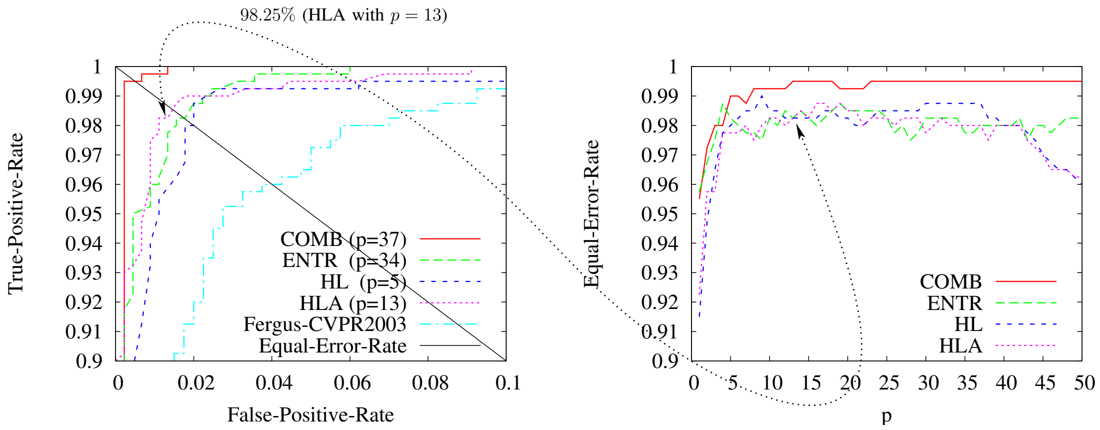


Fig. 8. Feature selection results with increasing n on a sample from the people database. This is one of the most challenging databases as the appearance of the people is very variable. In this case likelihood and mutual information focused on different *part classifiers*, there were no "very special" or "very general" clusters.

B. *Image classification*

The following experiments test for the presence of an object class in the given images. This evaluation criteria was chosen because the ground truth is clear, so the problem is well defined and easier to compare. The reports of other groups [11], [15] using the same

[4] We call a *part classifier* specific when it has a very high presision with a low recall rate. Corresponding parts appear rarely on the object but almost never on the background.



(a) (b)

Fig. 9. On the left, the ROC curves for image classification on the motorbikes database using different detectors and estimated p parameters. On the right the corresponding equal error rate curves. The dotted line with arrows shows the connection between the two curves. See the text for an explanation.

datasets were based on the same criteria. Receiver Operating Characteristic (ROC) curves are the most common way to report the efficiency of classifiers. They show correct detections as a function of incorrect detections. There are several ways to compare two ROC curves. Typically a specified operating point is chosen depending on the goal of the application. Here we chose the equal error rate, i.e. when the rate of true positives and true negatives are equal.

$$p(\text{True Positive}) = 1 - p(\text{False Positive}). \tag{17}$$

Fig. 9 shows an example of locating that point on the ROC curve. On Fig. 9 (a), besides the five ROC curves there is also a diagonal line labeled “Equal-Error-Rate”. Our chosen operating point is the highest point on the ROC curve below (or on) this diagonal line. For illustration on the ROC curve labeled as HLA the described point is singled out by an arrow.

In Section III-D we introduced p as a parameter of our final classifier, specifying the minimum number of positive detections on an image required to declare the presence of an object. We estimated p by maximizing (17) on the *validation set*.

Fig. 9 (b) shows a curve of equal error rate as a function of p. In our results the maximum of this function is selected the *ideal* p(of the given test set) allowing us to measure the performance of the combined *part classifiers* independently of the estimation p.

In Table I we present the results achieved with the combination of HL and ENTR. In this table all figures are reported on the test sets and with likelihood ($R_{\mathcal{L}}$) ranking. *Ideal p* indicates the performance with the ideally chosen p, while *estimated p* are results realized

TABLE I

EQUAL-ERROR-RATE RESULTS ON IMAGE CLASSIFICATION USING THE COMBINATION OF HL AND ENTR DETECTORS (COMB) AND $R_{\mathcal{L}}$ RANKING. THE LAST COLUMN SHOWS THE BEST RESULTS REPORTED BY *other groups* ON THE SAME DATASETS.

Database	This paper				Others
	Ideal p		Estimated p		
	p	%	p	%	%
Airplanes 	25	**98.75**	28	98.5	94.0 [11]
Faces 	45	**99.54**	33	99.08	96.8 [11]
Motorbikes 	37	**99.5**	37	**99.5**	96.0 [11]
Wild Cats 	7	**91.0**	13	87.0	90.0 [11]
Leaves 	8	**98.92**	8	**98.92**	84 [27]
Bikes 	26	**92.0**	14	88.0	86.5 [15]
People 	13	**88.0**	13	**88.0**	80.8 [15]

when the parameter p was learned on the validation set. In the last column, the rates of *other groups* using the same datasets are shown for comparison. The combination with the ideally chosen p always outperforms existing methods. Our estimation of this parameter was also particularly good, several times leading to ideal results, and otherwise to only a small drop in performance. There was only one database, the wild cats, when our COMB results with an estimated p performed worse than [11]. The individual performances of the detectors are summarized in Table II. The results realized from the output of single detectors are comparable, and most of the time better than existing methods Below we evaluate the results of different interest point detectors. HL performed better than ENTR for airplanes, faces, motorbikes and wild cats, while in case of leaves, bicycles and people the reverse was true. So there is no winner between the two. However some interesting facts are worth

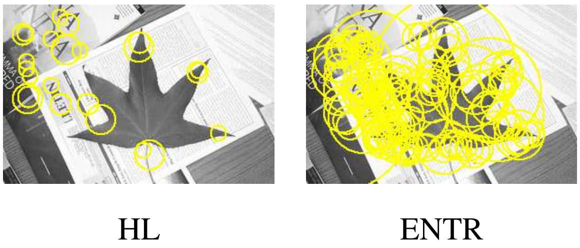


Fig. 10. The output of the HL and ENTR operators on the leaves database.

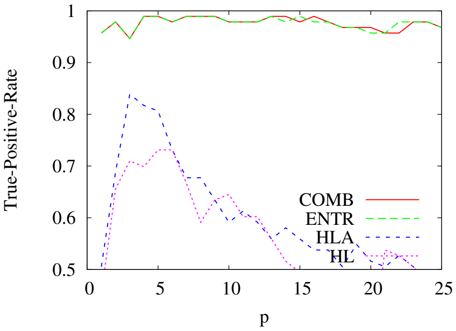


Fig. 11. Equal-error-rate results of image classification on the leaves database.

mentioning. HL performed very poorly (73%) on the leaves database. This is due to the fact that the detector itself performed very poorly on the object class: few HL points were found on the leaves, and due to the nature of corner detection and the structure of the leaves, most of the *object features* contained a huge amount of background. See Fig. 10 for an example result of the HL detector compared to ENTR. Fig. 11 shows the equal error rate, curve as a function of p. The curve of HL never reaches 75% and starts collapsing after $p = 5$. HLA, the affine invariant version of the same detector, performed relatively well with very few parts ($p = 3$) because the corresponding regions of the extracted object parts contained less background owing to their affine (ellipse) adaptations. HLA curve peaked at 83.9%, but the poor detection count caused high instability with changing p. E.g with $p = 2$ the result only 68.8%, and increasing p beyonds caused a similar drop.

The most challenging datasets were the people and bicycles. The changes in viewpoint and scale were relatively large compared to the other test sets, as were the changes in appearance of the people due to pose and clothing. In the case of the bicycles, the reported results are surprisingly good, with the exception of HLA. The poor performance with affine adapted regions was due to the structure of the objects — even when the corner detection correctly localized some significant parts, the affine estimation adjusted the ellipse on the background

TABLE II

EQUAL-ERROR-RATE RESULTS ON IMAGE CLASSIFICATION WITH DIFFERENT DATABASES, DETECTORS.

Database	Detector	Ideal p		Estimated p		Others
		p	%	p	%	%
Airplanes	ENTR	18	97.0	8	96.00	94.0
	HL	14	97.75	9	96.25	
	HLA	8	96.75	8	96.75	
Faces	ENTR	12	97.70	19	96.77	96.8
	HL	11	99.54	11	99.54	
	HLA	21	**100.0**	21	**100.0**	
Motorbikes	ENTR	4	98.75	11	98.0	96.0
	HL	9	99.0	5	98.0	
	HLA	16	98.75	13	98.25	
Wild Cats	ENTR	7	83.0	25	82.0	90.0
	HL	12	**93.0**	10	91.0	
	HLA	12	**92.0**	68	89.0	
Leaves	ENTR	8	**98.92**	8	**98.92**	84
	HL	5	73.12	2	65.59	
	HLA	3	83.87	2	68.82	
Bikes	ENTR	29	**92.0**	19	90.0	
	HL	24	84.0	24	84.0	
	HLA	32	70.0	12	64.0	86.5
People	ENTR	12	**88.0**	29	80.0	
	HL	27	78.0	30	76.0	
	HLA	21	76.0	17	74.0	80.8

between the spokes or on rich texture just next to the tire or other tubular parts. With the people database HL and HLA detections correctly determine significant object parts, but the detected regions were mostly on the boundary between the people and the background. Since our representation is based on the full patches, most of the object descriptors were contaminated by background textures. For such parts, the learning stage cannot generalize well, therefore the constructed *part classifiers* are not discriminative enough. The ENTR detector located more points on the people, leading to more discriminative *part classifiers*, and thus a 10% improvement in the classification rate. On the bicycles dataset we believe that the ENTR is better, owing to its good detection of a very discriminative part, the tire. Fig. 12 demonstrates that ENTR detects a large number of regions aligned around the tire.

Fig. 13 shows a typical image from the motorbike dataset where the output of the different

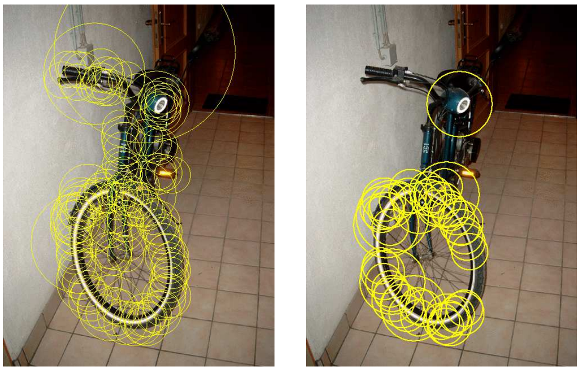


Fig. 12. Selection results on the bicycle database. The ENTR detector output is shown on the left, and the selected discriminative features are shown on the right.

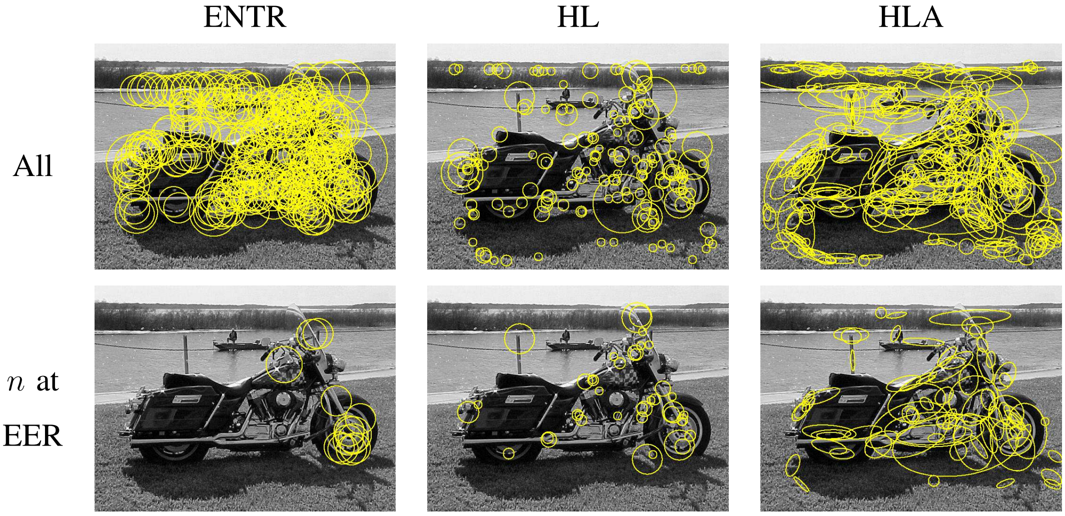


Fig. 13. Selection results using different feature detectors: Entropy of region histograms (ENTR) [8], Harris-Laplace (HL) [19], Harris-Affine (HLA) [20]. The top row shows the output of the interest point detectors, i.e the input to our selection method. In the bottom row we mark only the n best ranked features. For this example we set our parameter n according to the equall error rate operating point from our ROC curves.

detectors and our corresponding selections can be visually compared. The top row displays all of the features extracted by the different interest point detectors, and the bottom row shows the corresponding outcome of our selection method.

In our experiments adding affine invariance seldom improved the results, and often made them worse. These datasets do not contain significant viewpoint changes so the fact that there is no significant performance gain with affine adaptation is not surprising. However, on the faces database our selection method with HLA resulted in a perfect classification. The reason is that the elliptical representation of local neighborhood led to a more precise representation

TABLE III

EQUAL-ERROR-RATE RESULTS ON IMAGE CLASSIFICATION USING LIKELIHOOD AND MUTUAL INFORMATION AS RANKING METHODS.

Database	$R_{\mathcal{L}}$		$R_{\mathcal{I}}$	
	p	%	p	%
Airplanes	25	$\mathbf{98.75}$	37	98.5
Faces	45	$\mathbf{99.54}$	16	$\mathbf{99.54}$
Motorbikes	37	$\mathbf{99.5}$	49	99.0
Wild Cats	7	$\mathbf{91.0}$	41	90.0
Leaves	8	$\mathbf{98.92}$	9	97.85
Bikes	26	$\mathbf{92.0}$	14	90.0
People	13	$\mathbf{88.0}$	12	82.0

of specific parts such as the eyes and the mouth.

The combination (COMB) worked almost as we expected. It produced improvements in two cases: motorbikes and airplanes. In these cases, HL and ENTR performed about equally well, and the combination of results led to even better performance. The right curve in Fig. 9 clearly illustrates the power of a good combination. One can use higher p values and the results are still high and stable, and the system shows reduced sensitivity to changes in p. Combinations can also provide useful protection against detectors that performing poorly on certain databases. Fig. 11 is an example of this with the leaves database. The COMB curve almost strictly follows the ENTR one and in Table II gives exactly the same results as would be expected. It is not inevitable that combining different cues always leads to better performance. Even though adding new cues provides more information, poor quality and additional noise can reduce the overall performance to something between the effectiveness of the individual ones. An example of this is the wild cats database, where the combination performed intermediately between ENTR and HL.

Using mutual information ($R_{\mathcal{I}}$) as the ranking criterion does not change our results significantly. Table III compares the two methods using the combined detectors and the ideally[5] chosen p in each case. The results obtained with the $R_{\mathcal{I}}$ ranking criteria for individual detectors give similar conclusions to $R_{\mathcal{L}}$, therefore we do not detail them individually. While in these experiments the likelihood still selected very "specialized" *part classifiers*,

[5] The ideal p was chosen in order to compare the ranking methods independently of the estimation of p

the parameter n is hidden and evidently lower in case of mutual information. The overall performance of $R_{\mathcal{L}}$ was always greater than or similar to $R_{\mathcal{I}}$.

V. Conclusion

In this paper, we have introduced a method for constructing *part classifier* corresponding to similar object parts or textures. The method is based on local descriptors, thus providing robustness to occlusion and cluttered backgrounds.

The local descriptors are partially labeled by marking their source images as positive or negative, so the final selection system is trained in a weakly-supervised fashion, while the learning of the parts (model estimation) is completely unsupervised. The learned parts and descriptors in both training and test images are invariant to illumination, scale and optionally to rotation and affine deformations. Alignment, normalization and pre-segmentation of the images are therefore not necessary.

Two different ranking techniques were compared for selecting discriminative parts and dominant textures of object classes. The comparison showed that likelihood is well suited for object recognition and detection, while mutual information is better suited for sparse representation and for focus of attention mechanisms, that is rapid localization based on a few classifiers.

The comparison of interest point detectors showed that both corner-like and blob-like features are valuable, and provide sufficient information for appearance based recognition. However for particular databases one can be better than the other. Corner-like detectors capture more valuable information on highly textured classes as they provide well localized features that lie fully on the object, while blob detectors are more suitable for objects built from homogeneous parts.

We showed how to combine the detector outputs by ranking discriminant features together, which not only made a choice between the detectors unnecessary, but also improved our recognition performance.

Experiments on seven different databases confirmed our expectations and proved that a simple appearance model can compete and most of the time outperform results obtained with more complicated, spatial models.

This paper has illustrated the importance of feature selection and shown good results using purely appearance based modeling. Our future work will include development the next stage of object localization by extending our learning phase to establish spatial constraints between the detected parts.

ACKNOWLEDGMENT

This work was supported by PASCAL and the European project LAVA. The authors are grateful for David Lowe and Krystian Mikolajczyk for their code for local descriptors and detectors. We would like to thank Roger Mohr, Bill Triggs, Guillaume Bouchard and Peter Carbonetto for comments.

REFERENCES

[1] S. Ullman, E. Sali, and M. Vidal-Naquet, "A fragment-based approach to object representation and classification," in *4th International Workshop on Visual Form, Capri, Italy*, May 2001.

[2] M. Weber, M. Welling, and P. Perona, "Unsupervised learning of models for recognition," in *Proceedings of the 6th European Conference on Computer Vision, Dublin, Ireland*, 2000, pp. 18–32.

[3] B. Leibe and B. Schiele, "Interleaved object categorization and segmentation," in *Proceedings of the 14th British Machine Vision Conference, Norwich, England*, September 2003.

[4] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii, USA*, vol. I, 2001, pp. 511–518.

[5] C. Papageorgiou and T. Poggio, "A trainable system for object detection," *International Journal of Computer Vision*, vol. 38, no. 1, pp. 15–33, 2000.

[6] S. Mahamud and M. Hebert, "The optimal distance measure for object detection," in *Proceedings of the Conference on Computer Vision and Pattern Recognition, Madison, Wisconsin, USA*, 2003.

[7] S. Agarwal and D. Roth, "Learning a sparse representation for object detection," in *Proceedings of the 7th European Conference on Computer Vision, Copenhagen, Denmark*, vol. IV, 2002, pp. 113–127.

[8] T. Kadir and M. Brady, "Scale, saliency and image description," *International Journal of Computer Vision*, vol. 45, no. 2, pp. 83–105, 2001.

[9] G. Dorko and C. Schmid, "Selection of scale-invariant parts for object class recognition," in *Proceedings of the 9th International Conference on Computer Vision, Nice, France*, vol. 1, 2003, pp. 634–640.

[10] K. Sung and T. Poggio, "Example-based learning for view-based human face detection," IEEE *Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 39–51, 1998.

[11] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *Proceedings of the Conference on Computer Vision and Pattern Recognition, Madison, Wisconsin, USA*, vol. II, 2003, pp. 264–271.

[12] S. Lazebnik, C. Schmid, and J. Ponce, "Affine-invariant local descriptors and neighborhood statistics for texture recognition," in *Proceedings of the 9th International Conference on Computer Vision, Nice, France*, vol. 1, 2003, pp. 649–655.

[13] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Proceedings of the 9th International Conference on Computer Vision, Nice, France*, 2003.

[14] X. Chen, L. Gu, S. Li, and H.-J. Zhang, "Learning representative local features for face detection," in *Proceedings of the Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii, USA*, vol. I, 2001, pp. 1126–1131.

[15] A. Opelt, M. Fussenegger, A. Pinz, and P. Auer, "Weak hypotheses and boosting for generic object detection and recognition," in *Proceedings of the 8th European Conference on Computer Vision, Prague, Czech Republic*, vol. II, 2004, pp. 71–84.

[16] Y. Amit and D. Geman, "A computational model for visual selection," *Neural Computation*, vol. 11, no. 7, pp. 1691–1715, 1999.

[17] T. Rikert, M. Jones, and P. Viola, "A cluster-based statistical model for object detection," in *Proceedings of the 7th International Conference on Computer Vision, Kerkyra, Greece*, 1999, pp. 1046–1053.

[18] C. Schmid, "Constructing models for content-based image retrieval," in *Proceedings of the Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii, USA*, 2001.

[19] K. Mikolajczyk and C. Schmid, "Indexing based on scale invariant interest points," in *Proceedings of the 8th International Conference on Computer Vision, Vancouver, Canada*, vol. 1, 2001, pp. 525–531.

[20] ——, "An affine invariant interest point detector," in *Proceedings of the 7th European Conference on Computer Vision, Copenhagen, Denmark*, vol. I, May 2002, pp. 128–142.

[21] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions," in *Proceedings of the 13th British Machine Vision Conference, Cardiff, England*, 2002, pp. 384–393.

[22] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[23] F. Jurie and C. Schmid, "Scale-invariant shape features for recognition of object categories," in *Proceedings of the Conference on Computer Vision and Pattern Recognition, Washington, DC, USA*, vol. II, 2004, pp. 90–96.

[24] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," in *Proceedings of the Conference on Computer Vision and Pattern Recognition, Madison, Wisconsin, USA*, vol. 2, June 2003, pp. 257–263.

[25] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[26] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. McGraw Hill, 1991.

[27] M. Weber, M. Welling, and P. Perona, "Towards automatic discovery of object categories," in *Proceedings of the Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina, USA*, 2000, p. 2101.