Electronic Letters on Computer Vision and Image Analysis 8(3):15-26, 2009

A Performance Evaluation of Exact and Approximate Match Kernels for Object Recognition

Barbara Caputo and Luo Jie

Idiap Research Institute
Centre du Parc, Rue Marconi 19
1920 Martigny, Switzerland

Received 31st July 2009; accepted 11th January 2010

Abstract

Local features have repeatedly shown their effectiveness for object recognition during the last years, and they have consequently become the preferred descriptor for this type of problems. The solution of the correspondence problem is traditionally approached with exact or approximate techniques. In this paper we are interested in methods that solve the correspondence problem via the definition of a kernel function that makes it possible to use local features as input to a support vector machine. We single out the match kernel, an exact approach, and the pyramid match kernel, that uses instead an approximate strategy. We present a thorough experimental evaluation of the two methods on three different databases. Results show that the exact method performs consistently better than the approximate one, especially for the object identification task, when training on a decreasing number of images. Based on these findings and on the computational cost of each approach, we suggest some criteria for choosing between the two kernels given the application at hand.

Key Words: object recognition, local features, kernel methods

1 Introduction

Since the seminal work of Lowe [17], local descriptors have become the feature of choice for recognition of visual patterns from still images. The basic idea is to represent an image with an unordered set of features, computed as follows: first, an interest point detector selects points in the image that are likely to have a high informative content (like corners, borders, etc). Then, a small patch is selected around each detected point, and a feature descriptor is computed on it. Thanks to a substantial research effort by the community, today several interest point detectors and descriptors are available [26, 12, 11]. They showed to have very good performance in many different applications, such as object recognition and categorization [7] [8], texture recognition [14], image and video retrieval [28] [22], robot localization and place recognition [27]. The power of these representations is that they are distinctive, robust to occlusion, invariant to transform, and do not require segmentation. These local descriptors could be represented as feature vector of high dimensionality for each interest regions, and the number of features depends on the image contents as well the choice of different parameters of detection.

Correspondence to: <bcaputo@idiap.ch>

Recommended for acceptance by <Manuele Bicego>
ELCVIA ISSN:1577-5097
Published by Computer Vision Center / Universitat Autònoma de Barcelona, Barcelona, Spain

Another very strong research trend in visual recognition during the last decade has been the use of sophisticated statistical learning methods, like Support Vector Machines (SVM) for object recognition and categorization [1, 3, 13, 15]. A key component of these classification algorithms is the need to compute similarity measures on a Hilbert space with the so called kernel function [29]. This condition, that implies the definition of a metric in such space, has kept separated these two research lines for several years.

Recently, a stream of works has proposed different kernel functions able to compute similarities between unordered set of features. The use of these functions within SVM-based algorithms makes it possible to exploit fully the potential of both methods, leading to state of the art results on several applications [13, 15]. Between the family of kernels for local features, we can single out two, that are the most representative of the mainstream approaches: the match kernel [30] that mimics an exact matching procedure, and the pyramid match kernel [10] that instead opts for an approximate matching procedure. In spite of some partial results reported in the literature [10], it is still missing a comprehensive evaluation of these two approaches for the visual recognition problem.

This paper presents a thorough experimental comparison between the match kernel and the pyramid match kernel for the problems of object identification and categorization. We performed experiments on 3 different databases, and we compared results also with two other exact and approximate matching approaches. Our experiments clearly show that the exact approaches in general, and the match kernel approach in particular, consistently achieve a better performance compared to the approximate methods. This is particularly true for the object identification scenario, when training is performed on a small amount of data. In this setting our experiments show quite clearly that the approximate approach suffers in terms of performance. Armed with the results of our experimental evaluation and with the knowledge of the computational cost of each method, we derive some suggestions for choosing the most suitable kernel for a given application. We are not aware of any previous extensive experimental study of these methods for the object recognition problem. We believe that these results will be a useful resource for the community.

The rest of the paper is organized as follows: section 2 describes the match kernel and discusses its properties. Section 3 reviews the pyramid match kernel. Section 4 describes in details the experimental evaluation done on the two methods, reports results and discusses their implications. The paper concludes with an overall discussion.

2 Exact recognition with local features: the match kernel

The first method to explicitly address the problem of solving the correspondence problem within the SVM framework was the match kernel [30]. The authors proposed to build a similarity measure that mimic a greedy matching procedure. Denote by $\mathcal{I} = \{\boldsymbol{I}_i\}_{i=1}^m$ a set of images and $\mathcal{L} = \{\boldsymbol{L}_i\}_{i=1}^m$ the corresponding set of local features, with $\boldsymbol{L}_i = \{\boldsymbol{l}_j(\boldsymbol{I}_i)\}_{j=1}^{n_i}, i = 1, \dots m$. For all $(\boldsymbol{L}_h, \boldsymbol{L}_k) \in \mathcal{L}$, the match kernel is defined as

$$K_L(\boldsymbol{L}_h, \boldsymbol{L}_k) = \frac{1}{2}\left[\hat{K}(\boldsymbol{L}_h, \boldsymbol{L}_k) + \hat{K}(\boldsymbol{L}_k, \boldsymbol{L}_h)\right] \tag{1}$$

with

$$\hat{K}(\boldsymbol{L}_h, \boldsymbol{L}_k) = \frac{1}{n_h}\sum_{i=1}^{n_h} \max_{j=1,\dots n_k} \{K_l(\boldsymbol{l}_i(\boldsymbol{L}_h), \boldsymbol{l}_j(\boldsymbol{L}_k))\},$$

and $K_l(\boldsymbol{l}_i, \boldsymbol{l}_j)$ a Mercer kernel. The choice of K_l depends on the local descriptor, and it is usually set by the user according to some prior knowledge [30]. The algorithm complexity for computing each entry of the kernel matrix is $O(dm^2)$, with d equals to the feature dimension and m equals to the maximum number of interest points.

In spite of the claim in [30] the match kernel does not satisfy the Mercer condition, thus it is not positive definite. To prove this point, it is sufficient to present a counter example. Consider the matrices G_1, G_2 and their max $G_3 = max(G_1, G_2)$ with eigenvalues $\lambda_1, \lambda_2, \lambda_3$ as follows:

$$G_1 = [+3 - 1 - 2; -1 + 3 + 3; -2 + 4 + 8]; \lambda_1 = [+10.45 + 2.31 + 1.24],$$

$$G_2 = [+8 + 4 - 2; +4 + 4 - 1; -2 + 1 + 1]; \lambda_2 = [+10.74 + 0.40 + 1.86],$$

$$G_3 = [+8 + 4 - 2; +4 + 4 + 3; -2 + 4 + 8]; \lambda_3 = [-0.39 + 10.39 + 10.00].$$

We see that G_1 and G_2 are two positive definite Gram matrices but their max G_3 is not. This shows that in general the match kernel is not a Mercer kernel. In spite of this drawback, this kernel has been extensively used for visual recognition, and has shown remarkable performances for object categorization [3], action classification [13] and indoor place recognition [18]. Besides performance, the real issue of non-Mercer kernels is that they do not guarantee that the SVM optimization problem is convex, thus the solution might not converge to the absolute minimum.

Boughorbel et al [2] introduced a new definition of kernel positiveness based on a statistical approach. Their definition is such that includes most of Mercer kernels, and it shows that matching kernels are statistically positive definite. The basic idea is to bound the probability that the Gram matrix of a given kernel violates the Mercer condition. A sufficient condition for the Gram matrix to be positive definite is to be diagonal dominant, namely

$$|G_{ii}| \geq \sum_{i \neq j} |G_{ij}|, \forall i$$

Starting from the McDiarmid inequality [19], Boughorbel et al introduced a modified version of the inequality that permits to enforce the kernel positiveness with high probability by tuning the kernel parameters, provided that the considered kernel satisfies the following conditions:

- the kernel is constant on the diagonal:
$$K_\sigma(x_i, x_i) = K;$$
- the kernel is positive and bounded:
$$0 \leq K_\sigma \leq c;$$
- the kernel vanishes when the parameters go to zero:
$$K_\sigma(x, y) \rightarrow 0, \sigma \rightarrow 0, \quad for \quad x \neq y$$

We now show experimentally that the match kernel satisfies these conditions, namely that there always exists a range of the kernel parameters where the corresponding Gram matrix is diagonally dominant. Our experimental evaluation is similar to that reported in [2], also showing that the match kernel satisfies the three conditions above.

We therefore run a set of experiments. We used the ETH-80 database [16], containing about 3280 images, and from each image we extracted on average 45 SIFT features. We sampled randomly 100 images from the database, we computed the Gram matrix of the match kernel on the corresponding features, and we computed its minimum eigenvalue. We considered a varying number of interest points per image ($N = 1, 2, 3, 4, 5$), randomly sampled. We considered also a varying dimension of the features ($M = 1, 5, 10, 20, 50, 128$) by truncating the 128 SIFT feature vector and taking the first M values. For each possible combination of N and M, we repeated the experiments 100,000 times, for a total of 3,000,000 evaluations. Results are reported in Figure 1. We see that all the minimum eigenvalues computed are greater or equal to zero. The zero values are approached for values of M below 5, and for values of N smaller or equal to 2. Both these values are well below the typical values in visual applications. These experiments thus confirm the theoretical analysis of Boughorbel et al, showing that the match kernel can be safely used in visual recognition applications.

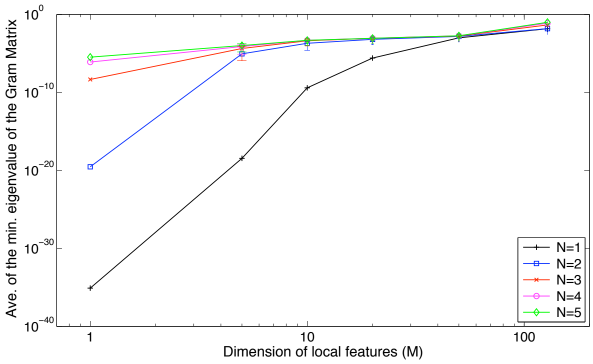


Figure 1: Average values of the minimum eigenvalue of the match kernel Gram matrix, for varying values of (M, N). Results were computed by an extensive simulation. The mean value is always greater of zero, and it approaches zero for values of (M, N) far below the typical values used in visual applications.

3 Approximate recognition with local features: the pyramid match kernel

An alternative approximate method was proposed by Grauman and Darrel [10]. Following the kernel-based approach, they introduced a kernel function that approximates the similarity measured by the optimal correspondence between feature sets of different dimension. This is achieved by mapping each feature to a multi-resolution histogram, preserving the individual feature's characteristic at the finest resolution level. The histograms are then compared using a linear combination of intersection kernels [10]. The resulting kernel is computationally very efficient, as it basically trades the exactness in matching features for a low algorithm complexity. In the rest of the section we will give the kernel definition, a sketch of the demonstration of its positive definitiveness, and its algorithmic complexity. For a more thorough discussion on the kernel we refer the reader to [10].

Let us consider a feature space F of d- dimensional vectors, and an input space S, containing sets of features drawn from F:

$$S = \{\boldsymbol{X} | \boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_m\}\}$$

with $\boldsymbol{x}_i \in F$..., and $m = |\boldsymbol{X}|$. The point dimension d is fixed for all features in F, but m can vary across instances in S. Let us also assume that the values of elements in vectors in F have a maximal range D. In order to approximate efficiently the optimal partial matching, Grauman and Darrel map each feature set to a multi-resolution histogram:

$$\Psi(\boldsymbol{X}) = [H_0(\boldsymbol{X}), \ldots, H_{L-1}(\boldsymbol{X}))]$$

where $\boldsymbol{X} \in S, L = [\log_2 D] + 1, H_i(\boldsymbol{X})$ is a histogram vector formed over points in $\boldsymbol{X}$ using a d-dimensional bins of side length 2^i, and $H_i(\boldsymbol{X})$ has dimension $r_i = (D/2^i)^d$. The pyramid match P_Δ measures similarity between point sets based on implicit correspondences found within the multi-resolution histogram space:

$$P_\Delta(\Psi(\boldsymbol{Y}), \Psi(\boldsymbol{Z})) = \sum_{i=0}^{L-1} w_i N_i \tag{2}$$

i.e. the similarity between two input sets $\boldsymbol{Y}$ and $\boldsymbol{Z}$ is defined as the weighted sum of the number of feature matchings found at each level of the pyramid formed by Ψ. N_i is computed using the histogram intersection function I:

$$I(\boldsymbol{A}, \boldsymbol{B}) = \sum_{j=1}^{r} min(\boldsymbol{A}^j, \boldsymbol{B}^j)$$

with $\boldsymbol{A}, \boldsymbol{B}$ histograms with r bins, and $\boldsymbol{A}^j$ denoting the count of the j^{th} bin of $\boldsymbol{A}$. N_i is given by

$$N_i = I(H_i(\boldsymbol{Y}), H_i(\boldsymbol{Z})) - I(H_{i-1}(\boldsymbol{Y}), H_{i-1}(\boldsymbol{Z}))$$

i.e. it is the difference between successive histogram levels' intersection, with H_i referring to the i^{th} component histogram generated by Ψ. The weighting w_i is given by $w_i = \frac{1}{2^i \cdot d}$. This choice implies that similarity at a finest resolution is rewarded more than similarity at a coarser level. Therefore, the kernel $P_\triangle$ can be written as

$$P_\triangle(\Psi(\boldsymbol{Y}), \Psi(\boldsymbol{Z})) = w_{L-1} I(H_{L-1}(\boldsymbol{Y}), H_{L-1}(\boldsymbol{Z})) + \sum_{i=0}^{L-2} (w_i - w_{i+1}) I(H_i(\boldsymbol{Y}), H_i(\boldsymbol{Z})). \tag{3}$$

$P_\triangle$ is a Mercer kernel. Indeed, the histogram intersection has been shown to be a Mercer kernel [1], and it is well known that a linear combination of Mercer kernels with positive coefficients is still a Mercer kernel [29]. Therefore, eq (3) is a Mercer kernel for any weighting scheme where $w_i \geq w_{i+1}$. Using the weights $w_i = \frac{1}{2^i \cdot d}$ ensures this property, thus ensuring also that $P_\triangle$ is a Mercer kernel.

An important property of this kernel is its lower algorithm complexity compared to the exact matching procedure. The time necessary to compute the L-level histogram pyramid $\Psi(\boldsymbol{X})$ for an input set with $x = |\boldsymbol{X}|$ d-dimensional features is $O(dzL)$, with $z = max(m, k)$ and k is the maximum histogram index value in a single dimension. The computational complexity of $P_\triangle$ is $O(dmL)$, as computing the intersection values for histograms that have been sorted by bin index requires a linear time in the number of non-zero entries. Finally, generating multiple pyramid matches with randomly shifted grid scales the complexity by T, the constant number of shifts (typically use $1 \leq T \leq 3$). Overall, the computational complexity of computing both the pyramids and the kernel is $O(TdmL)$. The L parameter can not be set to an arbitrary small value, usually L equals $\lceil log_2 D \rceil + 1$, where D is the value of the maximal feature range [10]. For example, for SIFT [17], $D \approx 250$, which yield $L = 9$. Therefore, the computational complexity is lower than the matching kernel when m is large.

The pyramid match kernel suffers from distortion factors that increase linearly with the dimension of the features [9]. To overcome this problem, Grauman and Darrell proposed a variant of the kernel that derives a hierarchical, data-dependent decomposition of the feature space that can be used to encode feature sets as multi-resolution histograms with non-uniformly shaped bins [9]. As for this kernel histogram pyramids are defined by the vocabulary, the authors call it the Vocabulary Guided- Pyramid Match Kernel (VG-PMK). The improved method is more accurate than old uniform bins methods. However, the computational cost also increased significantly. To generate the structure of the VG-pyramid, the algorithms have to perform hierarchical k-means clustering on some example feature vectors. The computational cost for generating this structure is $O(ILkdn)$, where I is the number of k-means iteration, L is the number of levels in the tree and n is the number of example feature vectors (typically it is required that $n \gg m$ in order to obtain a reasonable representation). This construction of the VG-pyramid structure is computationally very costly. Once the algorithm has constructed a VG-pyramid, the time necessary to embed an input set into the pyramid is O(kdmL), and the time required to compute a matching between two computed pyramids is $O(mL)$. We will use this version of the kernel in our object identification experiments when low dimensional features (e.g. PCA-SIFT) produce bad results.

4 Experiments

The previous sections described the exact and approximate match kernels, and showed that the approximate approach of Grauman and Darrell has a lower algorithm complexity than the exact match kernel. This is achieved by using an approximation when matching the local features. In this section we present experiments that test if using the approximate approach might lead to a decrease in performance, compared to the exact method, on object recognition problems. We focus specifically on the tasks of object identification and object categorization, running experiments on three publicly available databases. To provide a more comprehensive evaluation, we also benchmark against two non-SVM based methods (one exact and one approximate) well known in the literature. For all the experiments, we used our extended version of the *libsvm* library [4] with a one-vs-all multi-class extension. SVM and kernel parameters were determined during training via cross validation. In the following we first describe the experimental settings (section 4.1). Section 4.2 reports the results obtained for the object identification task, and section 4.3 reports our findings for the object categorization task.

4.1 Experimental Setup

In this section we first describe the databases used (section 4.1.1), then the feature representations (section 4.1.2) and finally we describe the two other baseline methods used in our experimental evaluation (section 4.1.3).

4.1.1 Databases

The ETH-80 database [16] consists of 80 objects from 8 different categories (apple, tomato, pear, toy-cows, toy-horses, toy-dogs, toy-cars and cups). Each object is represented by 41 images from viewpoints spaced equally over the upper viewing hemisphere, at distances of 22.5° – 26°. Objects are shown on a blue background without rescaling. Figure 2 presents one sample image per each object. We used this database for the object identification and object categorization experiments (section 4.2 and 4.3 respectively). For the object identification experiments, we considered separately all 80 objects: the training set consisted of five views per object, equally spaced on the viewing hemisphere. All the remaining images were used for testing. For the object categorization experiments, we considered 8 classes, one for each category: the training set consisted of five views per object, equally spaced, for 79 objects. The test set therefore contains all five images of one object, and the prediction performance is averaged over all 80 possible combinations of training and test set. This setup has been used first in [5] for object categorization experiments on the ETH-80 database. It is of interest for us here because it allows us to study the behavior of the two approaches when learning from small samples.

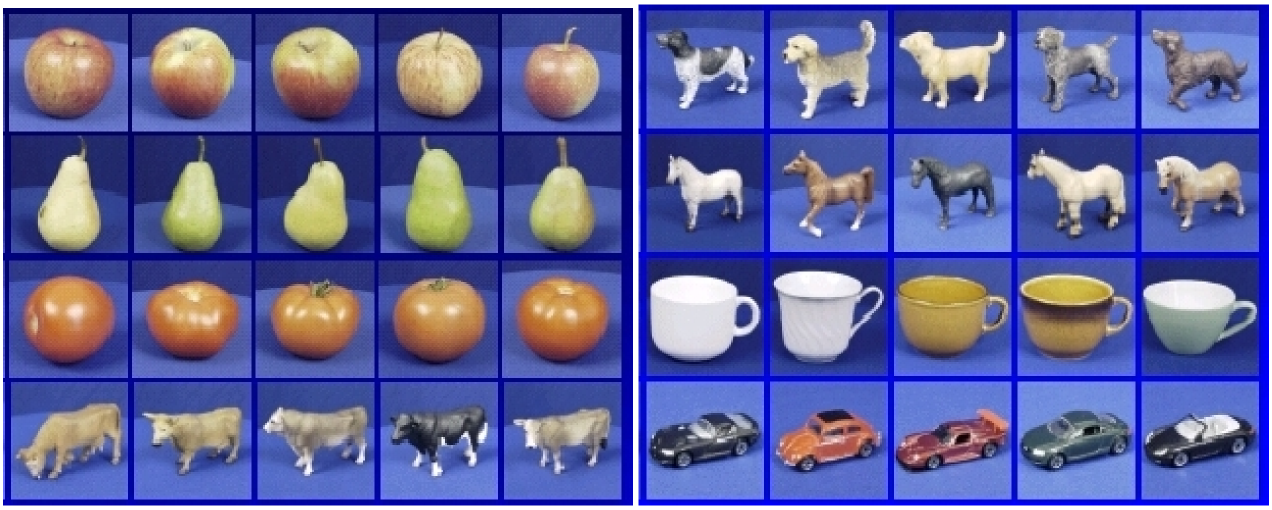
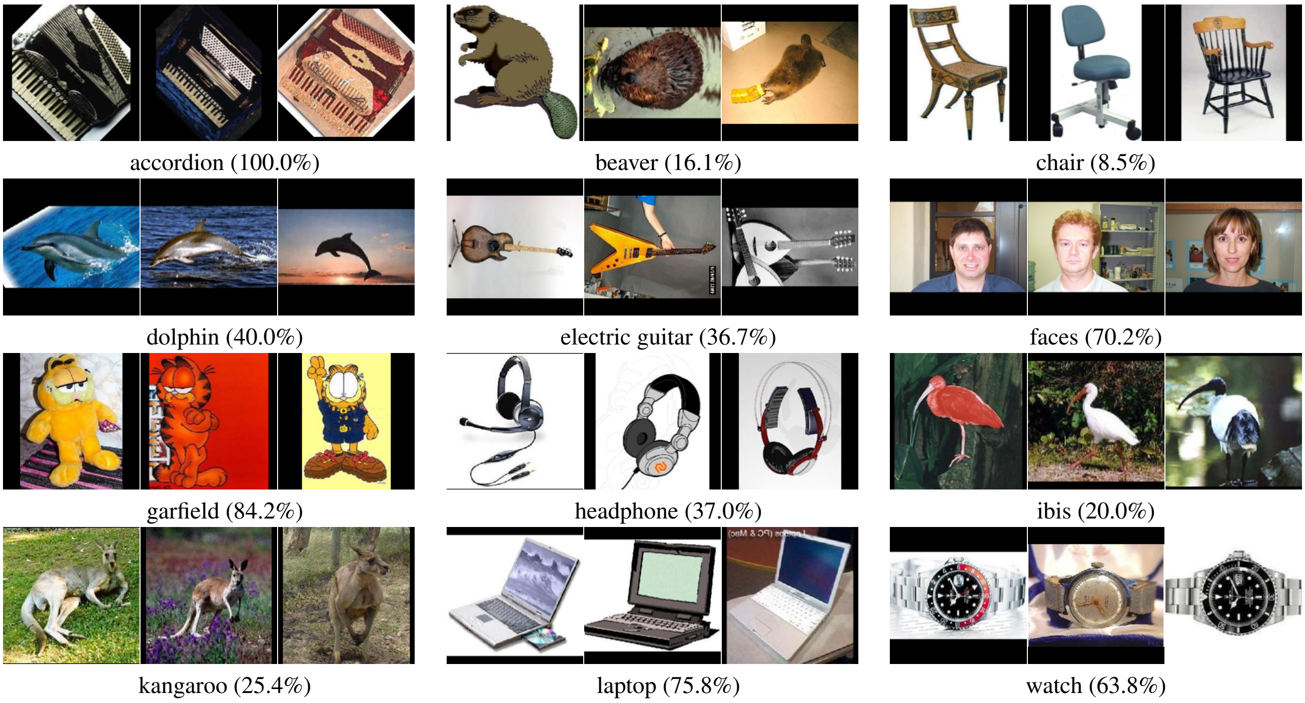


Figure 2: Example images from the ETH-80 objects database. For each category, we show images of five objects.

The Caltech-101 database [6] is a very popular benchmark for object categorization. It is a challenging database consisting of 101 object categories. There are 8677 images in the dataset, variously distributed across the different categories (from a minimum of 31 to a maximum of 800 images per category). Sample images for some of the categories are shown in Figure 3. The database was collected using the Google Image Search engine. As a consequence, many of the images contain a significant amount of intra-class variability. Note that most of the images in the database contain little or no clutter, and objects tend to lie in the center of the image. Furthermore, categories such as motorbike, airplane, cannon, etc. were manually flipped, so that all instances face the same direction. Categories with a predominantly vertical structure were rotated to an arbitrary angle, thus have a partially black background.

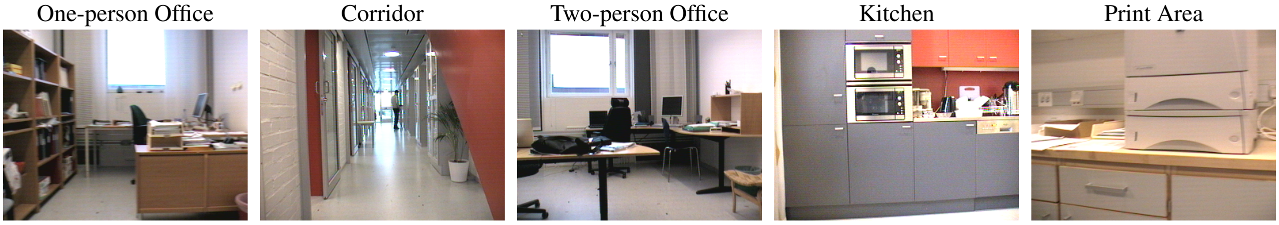
The KTH-IDOL2 database [18] is a place recognition database containing 24 image sequences acquired by a perspective camera, mounted on two mobile robot platforms. The acquisition was performed within an



accordion (100.0%)
beaver (16.1%)
chair (8.5%)
dolphin (40.0%)
electric guitar (36.7%)
faces (70.2%)
garfield (84.2%)
headphone (37.0%)
ibis (20.0%)
kangaroo (25.4%)
laptop (75.8%)
watch (63.8%)

Figure 3: Example images from the Caltech-101 database. Three images are shown for 12 of the 101 categories; we also report the classification rate obtained by the match kernel.

indoor environment consisting of five rooms with different functionality (one-person office, two-person office, corridor, kitchen and printer area) under various illumination conditions (cloudy weather, sunny weather, and night) across a span of six months. The robots were manually driven through an indoor laboratory environment and images were acquired at a rate of 5-fps. Each image sequence consists of 800 – 1100 frames automatically labeled with one of five different classes. Therefore, the data captures the natural variability that occurs in real-world environments introduced by both illumination and human activity (people appear in the rooms, furniture and objects change location etc.) Figure 4 shows some sample images from the database. As the focus of our work is not to study how to cope with long time variations and illumination differences, we used only the first part of the database for evaluations, and recognition experiments (train and test) are performed only on pairs of sequences acquired under stable illumination conditions and using the same robot. Note that, although the illumination conditions for both training and test images are very similar, the algorithm still has to tackle other kinds of variability such as viewpoint changes and presence/absence of people.



One-person Office
Corridor
Two-person Office
Kitchen
Print Area

Figure 4: Example images taken from the KTH-IDOL2 database showing the five rooms.

4.1.2 Feature representation

It has been shown that local affine- or scale- invariant feature descriptors extracted from a sparse set of interest points of an image have good performance in recognition tasks [17, 20]. Their success is mainly due to their distinctive representation, and their robustness to occlusion and affine transformations. Local feature extraction consists of two steps: first, an *interest point detector* finds interest points in the image; second, a local descriptor computes a feature vector from the image region localized at the interest points.

In all our experiments, we use the *Harris-Laplace detector* [21] and SIFT descriptor [17]. The Harris-Laplace detector responds to corner-like structures, and is invariant to rotation and scale transformations. The points are detected by the scale-adapted Harris function and selected in scale-space by the Laplacian-of-Gaussian operator. The number of detected points can be adjusted by a threshold parameter. The SIFT descriptor represents the features of local patches characterized by coordinates in the scale-space in the form of histograms of gradient direction. The descriptor vector is of dimension 128. SIFTs are invariant to image scaling, translation, rotation and are partially invariant to changing viewpoints and changes in illumination.

4.1.3 Baseline comparison approaches

Keypoint match[17] is performed by first matching each keypoint independently to the database of keypoints extracted from training images. The best candidate match for each keypoint is found by identifying its nearest neighbor in the database of keypoints from training images. Here, the nearest neighbor is defined as the keypoint with the minimum Euclidean distance to the matching one. However, due to ambiguous features or features arising from background clutter, there will be mismatches, as well as missing matches. To discard the ambiguous features, Lowe proposed an effective measure by comparing the distance of the closest neighbor to that of the second-closest neighbor. Plus, the second-closest neighbor is defined as the closest neighbor that comes from a different object than the first match. This measure is based on the assumption that correct matches need to have the closest neighbor significantly closer than the closest incorrect match to achieve reliable matching. In the original literature, the keypoint match implementation rejects all matches in which the distance ratio of the first and second neighbor is greater than 0.8. This value eliminates 90% of the false matches while discarding less than 5% of the correct matches. In our experiments, we adopt the same ratio value.

Bag-of-words first, a large number of SIFT features, computed from the sampled patches using a keypoint detector, are extracted from the image dataset. To build the "codebook" and classify the image, we use a similar implementation as the method presented in [22]. All the descriptors are hierarchically quantized in a vocabulary tree, which defines a hierarchical quantization that is built by hierarchical k-means clustering. A hierarchical scoring scheme is applied to recognition images from the training data set. As indicated in [22], better performance quality could be obtained with a larger vocabulary. Here, we build a vocabulary tree with ten branch factor and four levels, which results in 10,000 leaf nodes in total. This setup has been shown to give good performance in preliminary experiments.

4.2 Evaluation results: object identification task

Evaluation on ETH-80. We report here a comparative evaluation between the match kernel, the keypoint match, the VG-PMK, and the bag-of-words methods on object identification using the ETH-80 database. We conducted two series of experiments. In the first, the task consisted of identifying an object between a group of 10 objects, all belonging to the same category (10 apples, 10 cars, etc). In the second, the task consisted of identifying an object between a group of 80 objects, i.e. the whole ETH database. Figure 5 shows the identification results on each object category as well as the whole database.

We see that the match kernel approach outperforms the other three methods on both series of experiments; the second best performance is achieved by the keypoint match method. Note that both are exact methods. The VG-PMK and bag-of-words methods achieve lower performances on this object identification task. This is probably due to their approximate matching approach, which may result in many incorrect matches on objects of high similarity. As the number of total classes grows (multi-categories case), the generalization advantage of SVM-based methods becomes more significant: the classification performance of the two kernel methods maintains the same level, while the performance of the other two decreases significantly. Note that chance performance would be just 1.25%.

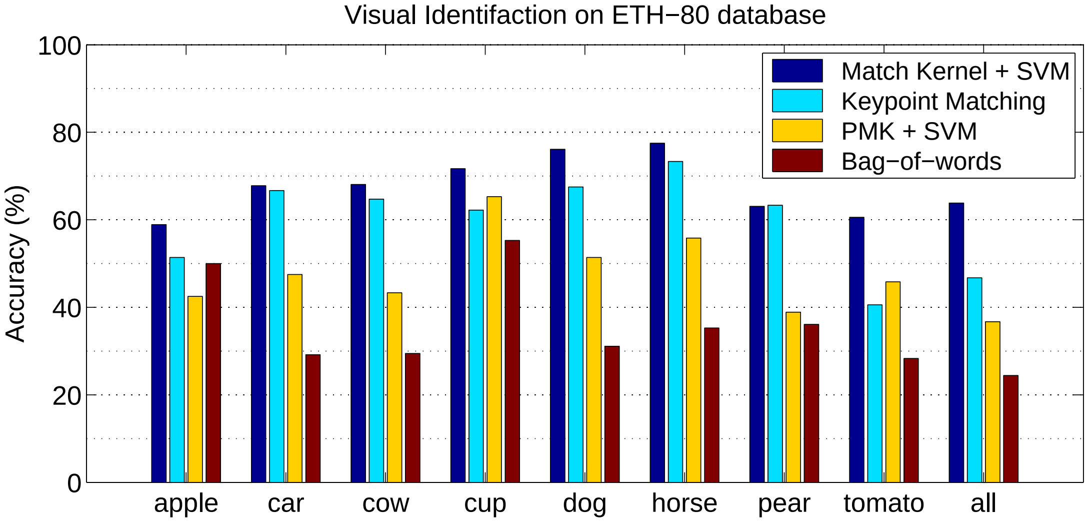


Figure 5: Comparison of different methods for objects identification on ETH-80 database.

Evaluation on KTH-IDOL2. As a second set of experiments on the object identification task, we evaluated the four approaches on a place recognition database. As the KTH-IDOL2 database contains six pairs of image sequences acquired by the same robot platforms under similar illumination and within close acquisition time, each classifier is trained on one sequence, and tested on the pair sequence. In total, twelve run of experiments were conducted; we report the average results with standard deviations (Table 4.2).

We see that overall the four methods achieve very high accuracy. The keypoint match method obtained the best result, closely followed by the match kernel approach. Both methods perform better than the bag-of-words and VG-PMK methods. It is worth noting that although the best result is obtained using the keypoint matching method, the searching is computationally expensive. To query each frame from a database containing 120,000 keypoints descriptors (which amounts to about 1,000 images) takes about 3 seconds on a 2.8GHZ Pentium IV machine with 2GB memory, which is not suitable for a robot localization task. This should be compared with the recognition time for the match kernel approach, that achieves the second best result, amounting to less than 300ms per frame in the same conditions. We can therefore conclude that the match kernel is the most effective recognition approach for the object identification task.

	Keypoint Match	Match Kernel + SVM	VG-PMK + SVM	Bag-of-words
Classification rate	97.44 ± 0.97	97.19 ± 1.61	94.53 ± 1.38	94.92 ± 1.57

Table 1: Classification rate for the place recognition task (KTH-IDOL database), for all methods.

4.3 Evaluation results: object categorization task

Evaluation on ETH-80. We report here the comparative evaluation between the match kernel, the VG-PMK and the bag of words methods on object categorization using the ETH80 database. We did not use for these experiments the keypoint match approach, because it is not suitable for this task. The experimental setup considered 8 object categories, and training and testing set were defined as described in section 4.1. Table 2 shows the obtained results for all methods. We see that the match kernel achieves the same performance as the bag-of-words approach. However, it is worth noting that the match kernel works on less than 10 percent of the features used by the bag-of-words approach [*]. It has been shown by many authors that recognition rates

[*] When extracting the features, we fixed the threshold of the Harris-Laplace detector so to produce an average of 45 interest points

always benefit from having larger numbers of features per images. In addition, when extracting an average of 256 features per set using uniformly sampling, the exact matching approach yields a recognition rate close to 90%, as reported in [10].

Evaluation on Caltech-101. For the Caltech-101 dataset, we followed the standard setup adopted in the literature, i.e. we randomly selected a certain number (15 in our experiment) of training images per class and test on the remaining images reporting the average accuracy. The recognition rates were normalized according to the number of test images per class. We repeated the random selection 10 times and report the average classification accuracy with standard deviations. The features were extracted using Harris-Laplace interest operator, and each image had on average 356 features. To built the bag-of-words representation we used the random sampling strategy suggest in [23], and found that 5,000 feature ("words") per image could obtain good performance. Table 3 shows the average classification results for all methods.

The best performance achieved with the bag-of-words approach on the Caltech-101 is of 39.36%, which is significantly lower than the 51% obtained by the match kernel. This result is slightly higher than the result obtained by the VG-PMK method (50%). We can also compare the result obtained by the match kernel with those reported in the literature for some other kernel approaches. For instance, the local kernel proposed by Zhang et al [31] obtains an accuracy of 53.9%, while the PMK built on sets of spatial features achieves an accuracy of 56.4% [15]. However, these results are built on more complicated or powerful representations, e.g., the PMK result is obtained by using features sampled uniformly on a grid which had on average 1140 features per images. This explains their slightly higher performance compared to the match kernel.

On the basis of these results, and of all the results reported in this paper, we can conclude that (a) exact matching is the strategy that yields the best results for the object recognition task, as it performs well for both object identification and object categorization; (b) the match kernel is the most effective exact approach, as it combines the power of SVM with the exact matching strategy. The gain in performance of the match kernel comes at a computational cost higher than that of VG-PMK. These points seem to suggest that whenever the computational cost during training is very relevant and when tackling the object categorization problem, one should opt for the approximate matching kernel. In all other cases, the exact matching kernel seems the best available option.

	Match Kernel + SVM	Bag-of-words	PMK[10]
Classification rate	75.5	75.5	73.0

Table 2: Comparison on object categorization on ETH-80 database.

	Match Kernel + SVM	Bag-of-words	PMK[10]
Classification rate	51.05	39.36	50

Table 3: Comparison on object categorization on Caltech-101 database.

5 Conclusions

This paper presents a benchmark evaluation between the match kernel and the pyramid match kernel, respectively an exact and an approximate approach for solving the correspondence problem between sets of unordered features. Our results show that the exact method is consistently better than the proximate one. A big selling point of approximate technique is, traditionally, their lower algorithm complexity compared to the exact methods. This should of course be considered when opting for a method or the other. Therefore, as the main

per image. As opposed to this, for sampling the patches from the images to build the codebook of the bag-of-words representation, we used the Lowe's keypoint detector, resulting in an average of 649 visual words per image.

contribution of this paper, we provide an experimentally grounded suggestion for when to favor the match kernel and when the VG-PMK.

While this paper aims at comparing two existing and popular kernels for local descriptors, it would be interesting to design new match kernels, based on matching heuristics that are commonly used for stereo image matching, such as [25, 24, 32]. The results shown in this paper should also be validated further, repeating the benckmark evaluation for several feature types, including space-time interest point detectors and descriptors [13].

References

[1] A. Barla, F. Odone, and A. Verri. Image kernels. In *Proc ICPR 2002*

[2] S. Boughorbel, J.-P. Tarel, and F. Fleuret. Non-mercer kernels for svm object recognition. In *Proc. BMVC'04*.

[3] B. Caputo, C. Wallraven, and M. E. Nilsback. Object categorization via local kernels. In *Proc. ICPR'04*.

[4] Chih Chung Chang and Chih Jen Lin. *LIBSVM: A Library for Support Vector Machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[5] J. Eichhorn and O. Chapelle. Object categorization with svm: kernels for local features. Technical Report 137, MPI for Biological Cybernetics, 2004.

[6] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):594–611, 2004.

[7] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proc CVPR03*.

[8] V. Ferrari, T. Tuytelaars, and L. Van Gool. Integrating multiple model views for object recognition. In *Proc CVPR04*.

[9] K. Grauman and T. Darrell. Approximate correspondences in high dimensions. In *Proc NIPS06*.

[10] K. Grauman and T. Darrell. The pyramid match kernel: Efficient learning with sets of features. *Journal of Machine Learning Research*, 8:725–760, 2007.

[11] A. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):433–449, 1999.

[12] Y. Ke and R Sukthankar. Pca-sift: A more distinctive representation for local image. In *Proc CVPR04*.

[13] I. Laptev, B. Caputo, T. Lindber, and C. Schultz. Local velocity-adapted motion events for spatio-temporal recognition. *Computer Vision and Image Understanding*, 108(3), 207-229, 2007.

[14] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1265–1278, 2005.

[15] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proc CVPR06*.

[16] B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *Proc CVPR03*.

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

[18] J. Luo, A. Pronobis, B. Caputo, and P. Jensfelt. Incremental learning for place recognition in dynamic environments. In *Proc. IROS'07*.

[19] C. McDiarmid. On the method of bounded differences. *London Mathematical Society Lecture Note Series*, 141(5), 1989.

[20] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2004.

[21] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.

[22] David Nister and Henrik Stewenius. Scalable recognition with a vocabulary tree. In *Proc CVPR06*.

[23] E. Nowak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. In *Proc ECCV06*.

[24] S. Pollard and John Mayhew and John Frisby. PMF: A stereo correspondence algorithm using a disparity gradient limit. In *Perception*, 14, 449–470, 1985.

[25] A. Rosenfeld and R. A. Hummel and S. Zucker.. Scene Labeling by Relaxation Operations In *IEEE Trans. Syst., Man and Cybern.*, 6, 420–433, 1976.

[26] C. Schmid and R. Mohr. Local greyvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–534, 1997.

[27] S. Se, D. Lowe, and J. Little. Vision-based mobile robot localization and mapping using scale-invariant features. In *Proc ICRA01*.

[28] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proc ICCV03*.

[29] V. Vapnik. *Statistical learning theory*. Wiley and Son, New York, 1998.

[30] C. Wallraven, B. Caputo, and A. Graf. Recognition with local features: the kernel recipe. In *Proc ICCV03*.

[31] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *International Journal of Computer Vision*, 73(2):213–238, 2007.

[32] S. W. Zucker and Y. G. Leclerc and J. L. Mohammed. Continuous Relaxation and Local Maxima Selection. In *Proc IJCAI79*.