International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol. 3, No. 1, March, 2010

Binarising SIFT-Descriptors to Reduce the Curse of Dimensionality in Histogram-Based Object Recognition

Martin Stommel

TZI Center for Computing and Communication Technologies
University Bremen, Am Fallturm 1, 28359 Bremen, Germany
mstommel@tzi.de

Abstract

It is shown that distance computations between SIFT-descriptors using the Euclidean distance suffer from the curse of dimensionality. The search for exact matches is less affected than the generalisation of image patterns, e.g. by clustering methods. Experimental results indicate that for the case of generalisation, the Hamming distance on binarised SIFT-descriptors is a much better choice. It is shown that the binary feature representation is visually plausible, numerically stable and information preserving. In an histogram-based object recognition system, the binary representation allows for the quick matching, compact storage and fast training of a code-book of features. A time-consuming clustering of the input data is redundant.

Keywords: *Object recognition, curse of dimensionality, SIFT, binarisation, clustering*

1. Introduction

Histogram-based methods have become increasingly popular for object recognition approaches that use a compositional model [17,20,23,5]. While in the initial Constellation Model [21] and related systems the appearance of the parts had been optimised individually [6,7], later systems (e.g. [17]) used generic code-books of frequently occuring patterns. The parts of a model are often represented conveniently by high-dimensional feature vectors, e.g. in the shape of SIFT-descriptors [15]. The code-book can then be created by clustering the input vectors [14]. The elements of the code-book represent the bins of a histogram of the feature space and allow for the measurement of class specific features frequencies.

Recent systems incorporate spatial [9,13] or class discriminatory [22] information, or focus on unsupervised training [7,11], speed [2,8], or the meaningful combination of different kinds of local descriptors [20].

However, the effects of the high-dimensionality of the part descriptors ("curse of dimensionality") on object recognition are usually ignored, except for unrelated or general studies [3,12]. Considering that popular descriptors have tens [10] or hundreds of dimensions and that negative effects have already been reported for as few as ten dimensions [3], this topic seems highly relevant.

Therefore, this paper addresses the dimensionality of the feature descriptors in the context of object recognition. For the popular example of SIFT-descriptors, the next section demonstrates that in fact there is a curse of dimensionality. Section 3 offers a possible solution which is supported by the experimental results given in the sections 4–6.

2. Distance between SIFT-descriptors

To perform experiments on the dimensionality of part descriptors, a data set of 200 images is sampled from ARD (a German TV network) TV news reviews of the year 2006. The images are sampled with a step width of 50 frames to ensure a minimum diversity. Then SIFT features are computed for these images with the default parameters given in the original paper [16], except for a higher contrast threshold of 10 and no initial doubling of the image size. Depending on the experiment, subsets with 100–500000 descriptors are selected randomly.

A prerequisite for histogram-based recognition methods is a distance measure that allows for the clustering of descriptors and the comparison of an image to the trained model. Lowe [15] proposes to compare SIFT-descriptors by the Euclidean distance. More generally, the distance between two descriptors *x*,*y* can be computed via the L_p-norm

$$L_p(x,y) = \left(\frac{1}{n}\sum_{i=1}^{n}|x_i - y_i|^p\right)^{\frac{1}{p}}, \quad x = (x_1, x_2, \ldots, x_n) \quad \text{etc.} \tag{1}$$

Beyer et al. [3] explain the curse of dimensionality as a general unreliability of high-dimensional distance computations that occurs if the distance distribution becomes sparse with rising dimensionality *n*. Close and distant points would approach each other, thus making it more difficult to distinguish between different objects.

To test if the curse of dimensionality affects the comparison of SIFT-descriptors, sets of 1000 descriptors are created. For each set, the dimensionality of the descriptors is set to

a certain value *n* by projecting the descriptors to a randomly chosen subspace. Then a full distance matrix is computed for each set.

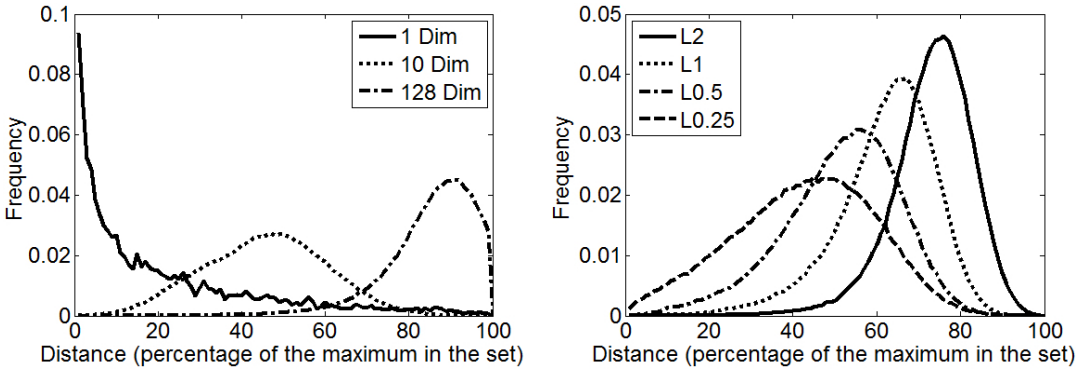

Figure 1. Distance distributions for different dimensionalities (left chart, using the Euclidean distance) and different distance norms (right chart, for a dimensionality of 90).

Figure 1 shows histograms over the distances for different choices of the dimensionality *n* and the norm *p* (Eq. 1). Distances of zero occur trivially very often, so they are not shown in the diagrams.

It can be seen that with increasing dimensionality the maximum of the distribution moves to the right, i.e. towards the maximum distance. The distances between any two elements becomes large and the distribution becomes sparse. This is exactly the effect reported by Beyer et al. Figure 1 (right) shows also that the effect is reduced by choosing a smaller, or even fractional norm as proposed by Aggarwal et al. [1]. However, the distance computation

becomes numerically delicate for the computation of high roots. In fact, it was not possible to achieve stable results for the L0.5-norm and descriptors with more than 90 elements. The roughness of the curve for the L0.25-norm in Fig. 1 (right) is a result of the beginning instability.

What does this mean for the similarity of image patterns? Figure 2 shows a list of image patterns arranged according to the Euclidean distance to a reference descriptor. Generally, it appears that the image patches are only similar for minimal distances. For medium and higher distances the Euclidean distance has no visual effect on the ordering

of the patterns. Also, some obviously similar patterns yield clearly different distances. However, one explanation for this may be that the SIFT operator creates multiple descriptors for a single feature if the orientation is ambiguous.

Generally, the Euclidean distance seems only appropriate for very small distances, i.e. the search for exact matches. For the goal of abstraction, a measure would be desirable that also allows for a gradual comparison of more dissimilar image regions. One possible solution is to reduce the dimensionality by sampling less than 8 gradient orientations or less than 16 grid cells in the image area. On the other hand, SIFT is generally a high dimensional method, so descriptors with less than 10 dimensions can not be expected.

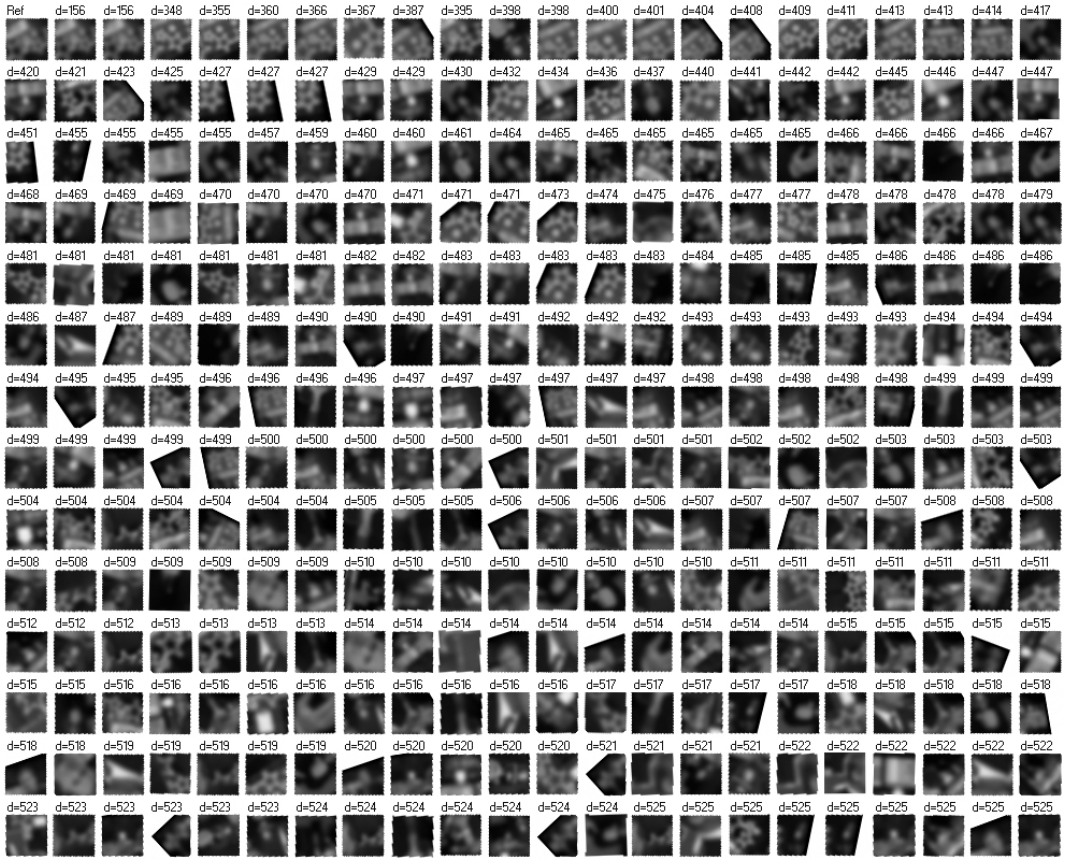

Figure 2. Image patches ordered by the Euclidean distance of their corresponding descriptor from a reference descriptor (first patch). The distance is written on top of each patch. Due to a (linear) scaling of the data, the maximum distance in the set is about 700.

3. Binarisation of feature descriptors

Alternatively to using a lower distance norm, Korn et al. [12] propose to reduce the dimensionality by the analysis of self-similarity of the distribution. However, a box-count plot of a set of 200000 descriptors (Fig. 3) does not indicate the required constant Hausdorff-dimension.

Other approaches include a feature selection or a feature transform to reduce the descriptor length [4,10]. While a certain reduction can be achieved by subsuming correlated descriptor components, all dimensions represent basically the same kind of information. Since minor differences are only introduced during the descriptor normalisation, the solution seems rather application dependent.

On the other hand, the box-count plot shows that even for a coarse quantisation with only two bins, i.e. a grid width of 128, almost the full diversity of the set is maintained. Since about 50% of the descriptor values are lower than twelve, this diversity must result from the few descriptor components with higher values. This observation suggests that a binarisation of the descriptor values, i.e. using only the values 0 and 1 for single descriptor components, would only minimally affect the object recognition. Though a binarisation does not reduce the dimensionality, the complexity of the problem would be reduced dramatically from 256^{128} different SIFT-descriptors to 2^{128} possibilities.

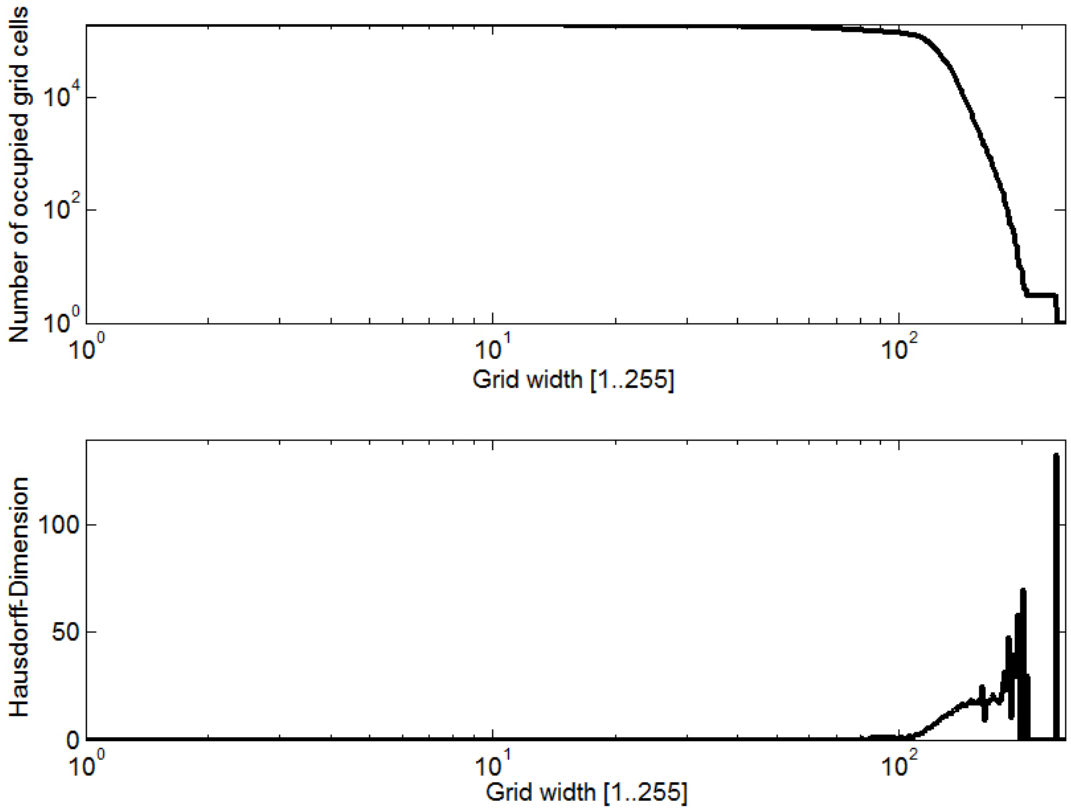

Figure 3. Box-count plot for a set of 200000 SIFT-descriptors.

The binarisation is carried out as follows: Each of the 128 values of a SIFT-descriptor is compared to a threshold. If the value is greater than the threshold, it is set to one. Otherwise it is set to zero.

At first, a threshold of 128 is chosen dividing the domain of descriptor values in the middle. The results are however bad: About 40 percent of all SIFT-descriptors are projected to the same binary descriptor. Obviously, the distribution is not centred to the middle of the domain of descriptor values. This agrees with the already mentioned predominance of small values.

To compensate for this, the mean values of every descriptor element are chosen as thresholds. There is one threshold for every descriptor component, i.e. 128 together. Still up to 4000 descriptors result in the same binary descriptor while the majority of binary descriptors corresponds to only one SIFT-descriptor in the original set. The mapping is thus very uneven, indicating that the distribution of SIFT-descriptors is not symmetrically centred around the mean.

To compensate for this asymmetry, the medians of each descriptor element are chosen as thresholds. Now, only 20 binary descriptors occur more than ten times. The binary descriptors are thus almost uniformly distributed. The remaining inaccuracies stem from the median computation. For an original set of 100000 SIFT-descriptors, this method results in 94376 binary descriptors. This number also includes SIFT-descriptors that have already been identical before binarisation. Since this concerns a proportion of 4.0 percent, only 1.6 percent of original SIFT-descriptors became indistinguishable because of the binarisation.

A natural way to compare binarised descriptors is to use the Hamming distance, i.e. the number of differing bits between two vectors.

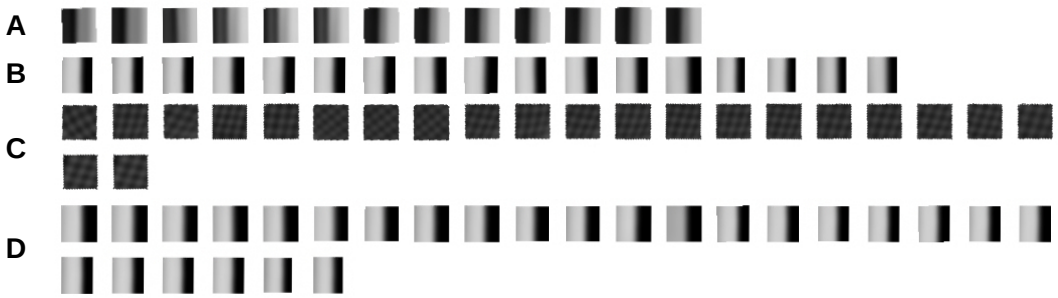

Figure 4. Patterns from the original images that are mapped to binary descriptors A–D. No distinction is made whether the patterns are combined because the original descriptors are identical or because the binarisation leads to identical results.

4. Correspondence between binarised descriptors and the image contents

The validation addresses first the information stored in the lost 1.6 percent of the descriptors. Is it possible that strongly differing patterns are spuriously combined by such a coarse quantisation? Figure 4 lists the patterns that are mapped to the same binary descriptor for four examples A–D. The groups are very homogeneous and it seems that no visually important effect is disregarded. It is in the contrary probable that different binary descriptors correspond to highly similar patterns, as for the examples B and D. For a recognition system

using a code-book of prototypical descriptor, these descriptors are potential candidates for a further combination.

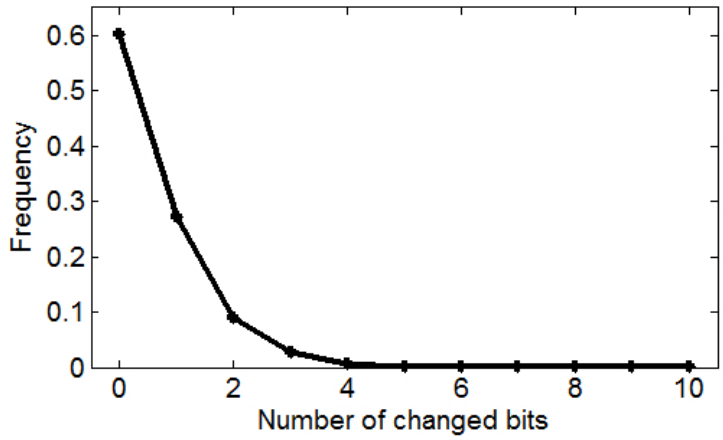

Figure 5. Histogram of the Hamming distances between differently binarised features. Only a small percentage of the bits varies due to different binarisation thresholds.

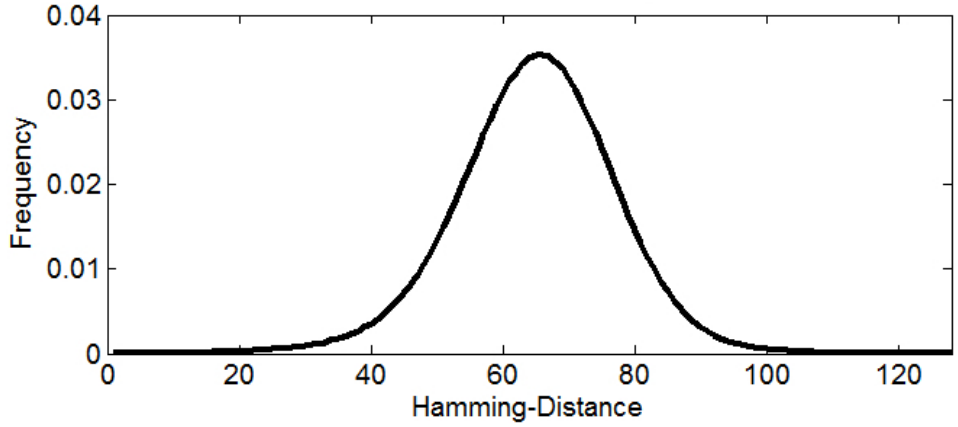

Figure 6. Hamming distances of binary descriptors

Since the Hamming distance depends directly on the binarisation thresholds, the stability of the median computation must be determined. To this end, the binarisation process is applied to 50 different subdivisions of the the Graz'02 image data base [18]. The data base consists of 460 images of persons, 373 images of bikes and 270 images without bikes or persons. For every subdivision, 130 images per class are selected randomly. The SIFT-descriptors and their medians are determined. We measured a standard deviation of 0.27 on average and 0.77 maximally (referring to an interval of [0 255] for the descriptor values) for medians of single descriptor components over the 50 subdivisions. The medians are now used to binarise another set of 100000 randomly chosen SIFT-descriptors taken from the whole data base. For every feature, the Hamming distances between all pairs of possible binarisations resulting from the 50 subdivisions are computed. Figure 5 shows the histogram of the outcome. For 60% of the features, no change occurs due to different binarisation thresholds. For 99% of the

features, at most 3 bits change. Obviously, the computation of the medians has only a very small influence on the binarisation. It can be regarded therefore as an independent step in the process chain.

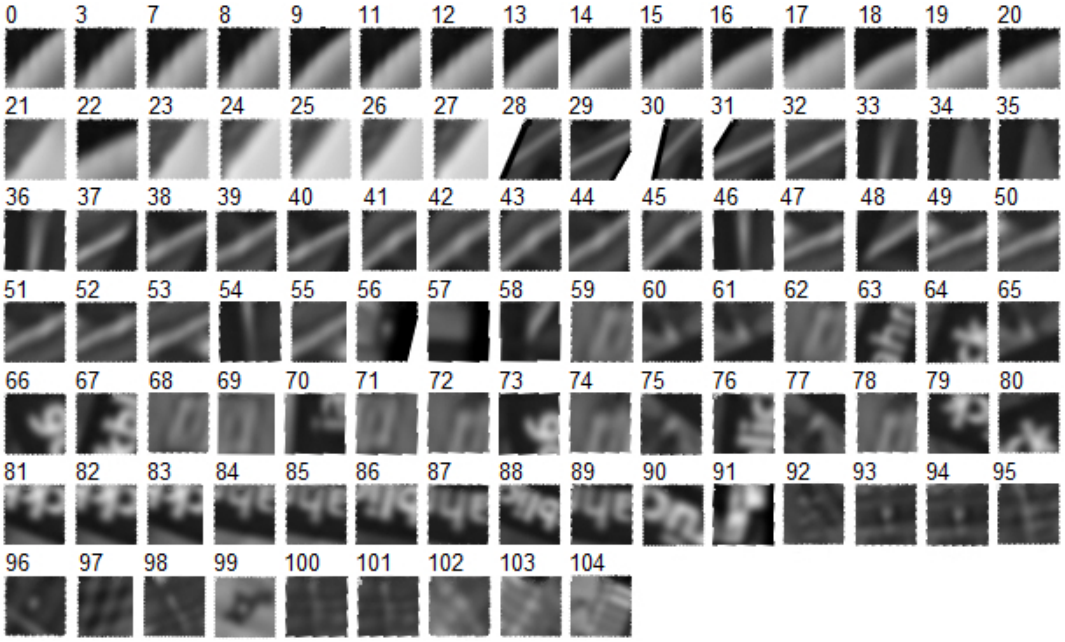

Figure 7. A subset of the patterns sorted by the Hamming distance to the first pattern (the upper left one). The numbers give the distances. For every distance only one pattern is shown. Distances greater than 104 did not occur in the set. In comparison to the Euclidean distance, the sorting by the Hamming distance shows a softer and more continuous increase in dissimilarity towards the maximum distance. Similar patterns cluster more obviously.

The next question is whether the Hamming distance is affected by the curse of dimensionality. To answer this question, 100000 binary descriptors are created using the component-wise medians as thresholds. Then the Hamming distances between all pairs of binary descriptors are calculated, resulting in 10^{10} distance values. Figure 6 shows a histogram of the distance values. Because of the descriptor length, the Hamming distances range from 0 to 128. The distance distribution is symmetric and centred around the value 64. There is no general shift towards high distance values ("sparseness"), so the curse of dimensionality does not seem to influence the descriptor comparison.

To exclude that the Hamming distance is an artificial measure without visual correspondence to the image data, it is important to visually compare image patches with low and high Hamming distances. Figure 7 shows a list of patterns and the Hamming distances of the corresponding descriptors. The first pattern in Fig. 7 is the reference pattern. The patterns are ordered by Hamming distance. In the case of multiple patterns with the same distance, one pattern is chosen randomly.

The figure shows a clear correspondence between distance and visual appearance. Even for higher distances, neighboring patterns are visually related but also different to patterns with another distance. Compared to the Euclidean distance (Fig. 2), the results for the Hamming

distance (Fig. 7) are obviously better. This makes the Hamming distance measure highly interesting for algorithms that aim at the generalisation of visually similar object parts.

5. Application to a visual code-book for object recognition

The use of the binary feature representation for histogram based object recognition (e.g. [14, 17, 21, 22]) is now studied in more detail. A resource-saving recognition system is introduced and the recognition rates are compared for binarised features and SIFT-descriptors.

In order to compute a histogram of high-dimensional descriptors, a code-book of sample features must be computed. The elements of the code-book represent the bins of the histogram in feature space. In order to keep the code-book at a feasible size, it is usually sparsely sampled from the feature space using heuristic learning algorithms [14,17,21,22]. In-between positions are often assigned to the most similar element, resulting in a discrete subdivision of the feature space.

However, the use of binarised features motivates two significant modifications: Firstly, because of the good generalisation properties of the Hamming distance to higher values, in-between positions are interpolated as linear combinations of the elements of the code-book. The code-book therefore resembles a coordinate system where the axes represent the distances of a feature to the elements. Secondly, a code-book of completely synthetic features is used, i.e. the code-book is not sampled from the input data but created independently. The advantage is that we can skip the extremely time-consuming clustering of a potentially large set of features and that we can easily adapt the statistical properties of the code-book to our needs.

Figure 8. Parts of faces extracted from the FERET data base. The image regions shown here correspond to the regions that SIFT-descriptors are computed for. The data base itself contains portraits of people together with annotations on the positions of the eyes, nose, and mouth. The annotations are used to localise the face features. The radii of the local regions are estimated from the extension of the eye, nose and mouth positions.

The exact procedure is as follows: First, a code-book of bit vectors is computed. The bits are randomly set with a probability of 0.5. The bit distribution is shaped as in the ideally independent case. The computational cost of this step is negligible. The use of bit vectors

also reduces the memory consumption by a factor of 256 compared to the original SIFT-descriptors. A descriptor of the input set is represented as a *distance vector* consisting of the Hamming distances to all elements of the code-book. Since the distance computations can be done very fast by using a look-up-table, the proposed method also facilitates the feature matching (a problem mentioned e.g. by Bay et al. [2]). To extract information about local image areas, the distance vectors can be classified. This is useful when a particular part of an object is being searched for.

A whole image is then represented as the *mean vector* of the distance vectors for all descriptors of that image. The mean vector therefore reflects the relative frequency of similarities between the image and certain elements of the code-book. The recognition of complex objects is done by classifying these mean vectors. The ability of the system to represent parts of objects on the one hand and classes of images on the other hand is now analysed in two experiments.

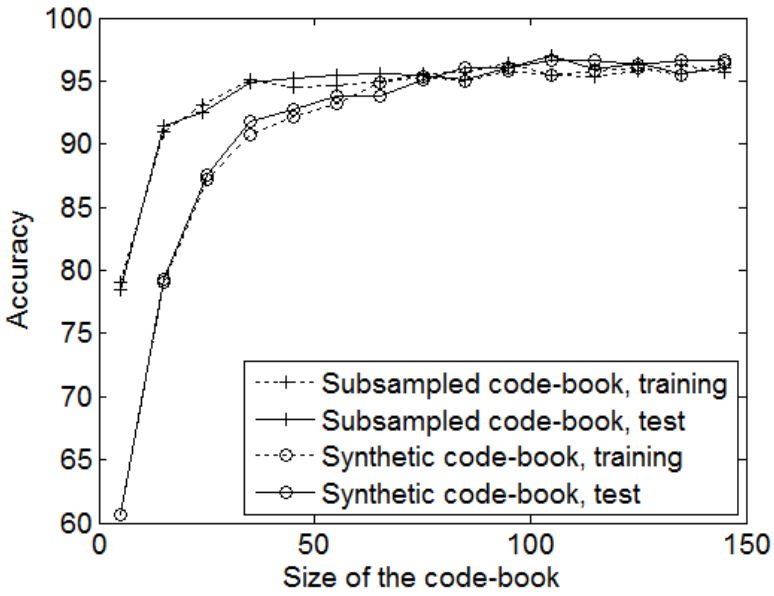

Figure 9. Size of the code-book plotted versus the accuracy of the classification.

To clarify the first point, it is tested if the distance vectors can be classified with the same accuracy as SIFT-descriptors. To this end, a set of 5000 SIFT-descriptors of 6 classes of face regions is collected, randomised and split into equally sized training and test sets. The 6 classes represent left eyes, right eyes, noses, mouths, whole faces, and non-face parts as a rejection class. The face parts are sampled from the Facial Recognition Technology (FERET) data base (http://www.itl.nist.gov/iad/humanid/feret/). The positions of the descriptors are taken from the annotation of the FERET data base. The radii of the descriptors are estimated from the extension of the part coordinates in the image area. Since the image background is usually homogeneous, the non-face parts are sampled from images of the GRAZ data base [18] that do not contain faces. To improve the robustness against small translations, the normalisation to a canonical feature orientation is omitted in this experiment. Figure 8 shows some of the face parts. Synthetic code-books of different size are created using the method described above and the distance vectors are computed for the corresponding part descriptors. For comparison, additional code-books are created by selecting random descriptors from the

binarised training set. A C-SVM classifier with Gaussian kernel is optimised first on the SIFT-descriptors from the training set and secondly on the distance vectors. The resulting models are then applied to the test set.

For the SIFT-descriptors, an accuracy of 98.8% on the training set and 98.7% on the test set is achieved. Figure 9 shows the resulting recognition rates for the code-book based classification. For smaller sizes, the subsampled code-books yield clearly better recognition rates than the completely synthetic code-books. This may indicate an advantage for clustering algorithms which assemble the code-book from the training samples. On the other hand, the difference disappears for code-books with more than 75 elements reaching an upper limit of approximately 96% accuracy. This is only 2.7% less than for the direct classification of SIFT-descriptors. Taking into account the enormous savings of CPU-time and memory using the synthetic code-book, the price/performance ratio of the method must be considered very good.

6. Image classification using a histogram of binarised descriptors

To measure the performance of the synthetic code-book for the recognition of bigger objects and full images, the method is now applied to the GRAZ'02 data base [18]. The image set is randomly split into equally sized training and test sets. For every image, the SIFT-features are computed binarised. The binarisation thresholds are computed as the medians over the descriptors of 100 images per class. A synthetic code-book of 1000 elements is created. This is to ensure that no information is lost because of a too small code-book. Usually, code-books with more than 100 elements seem uncritical in this respect. Then for every image, the mean vector is computed as described above. The mean vectors of the training set are used as the prototypes for a k-nearest neighbour classifier. However, the notion of a prototype for a background image may be a bit misleading because the background images do not represent particular objects. Consequently, the distribution of the background vectors overlaps with the other classes and does not form an equally clear cluster. To compensate for this, the optimal k-neighbourhood composition is estimated additionally to the value *k* of the neighbourhood size during training [19].

The system achieves an accuracy of 76.7% on the test images. Averaged over all classes (person, bike, rejection), a recall of 79% and a precision of 80% is achieved. Table 1 gives the recognition rates separated by class. The results bear comparison with the literature: Using the same type of base feature, the authors of the data base report a markedly lower precision between 70% and 64% for a boosting approach while a similar recall of 78–79% is achieved. Although a direct comparison of the results is only possible to a certain degree, the synthetic alphabet seems to preserve the information of the local features much better.

Table 1. Test results on the GRAZ data set

		Real class		
		Persons	Bikes	Background
Recognised class	Persons	74%	20%	3%
	Bikes	2%	65%	0%
	Background	24%	16%	97%

Figure 10. Samples from the Graz data base. The rows show four examples for the person, bike, and background class.

7. Conclusion

The experiments show that the comparison of SIFT-descriptors by the Euclidean distance suffers from the curse of dimensionality. Although this does not seem to substantially influence the search for exact matches, it makes a gradual comparison of more dissimilar image regions unreliable. The gradual comparison however is a prerequisite for many modern compositional object recognition techniques. To solve this problem, binarised descriptors together with the Hamming distance are proposed. The experiments show that only 1.6% of the original information is lost by the binarisation. Compared to the Euclidean distance, the proposed method is visually more plausible and does not show the sparseness of the distance distribution seen for the Euclidean distance. The computation of the binarisation thresholds is robust and can be conducted widely independently of subsequent calculations. The practical application to a histogram-based image classification task reveals more benefits: The time-consuming clustering of the input data can be omitted. Instead, a completely synthetic code-book of compact bit-vectors can be generated at negligible cost. The feature matching can be conducted extremely fast using precomputed look-up-tables. The loss of information introduced by the code-book representation is only 2–3% compared to the original SIFT-descriptors. Competitive object recognition results on the GRAZ data set support the quality of the proposed method.

References

[1] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, On the Surprising Behavior of Distance Metrics in High Dimensional Space, ICDT 2001, pp. 420–434.

[2] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, SURF: Speeded Up Robust Features, Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, 2008, pp. 346–359.

[3] K. Beyer, J.Goldstein, R. Ramakrishnan, and U. Shaft, When Is "Nearest Neighbor" Meaningful?, In Int. Conf. on Database Theory, 1999, pp. 217–235.

[4] B. Bonev, F. Escolano, and M. Cazorla, Feature selection, mutual information, and the classification of high-dimensional patterns, Pattern Analysis and Applications, February 2008, pp. 309–319.

[5] A. Bosch, A. Zisserman, and X. Munoz, Scene Classification Using a Hybrid Generative/Discriminative Approach, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 30(4), 2008, pp. 712–227.

[6] D. J. Crandall, P. F. Felzenszwalb, and D. P. Huttenlocher, Spatial Priors for Part-Based Recognition Using Statistical Models. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Vol. 1, 2005, pp. 10–17.

[7] R. Fergus, P. Perona, and A. Zisserman. Weakly Supervised Scale-Invariant Learning of Models for Visual Recognition. In International Journal of Computer Vision, volume 71, March 2007, pp. 273–303.

[8] M. Grabner, H. Grabner, and H. Bischof, Fast Approximated SIFT, Asian Conference on Computer Vision (ACCV), 2006, pp. 918–927.

[9] K. Grauman and T. Darrell, The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features, In proc. IEEE International Conference on Computer Vision (ICCV), Vol. 2, 2005, pp. 1458–1465.

[10] Y. Ke, R. Sukthankar, PCA-SIFT: A More Distinctive Representation for Local Image Descriptors, In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, 2004, pp. 506–513.

[11] G. Kim, C. Faloutsos, and M. Herbert, Unsupervised Modeling of Object Categories Using Link Analysis Techniques, In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008.

[12] F. Korn, B.-U. Pagel, and C. Faloutsos, On the "Dimensionality Curse" and the "Self-Similarity Blessing", IEEE Transactions on Knowledge and Data Engineering, Vol. 13(1), 2001, pp. 96–111.

[13] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Vol. 2, 2006, pp. 2169–2178.

[14] B. Leibe, K. Mikolajczyk, and B. Schiele, Efficient clustering and matching for object class recognition. Proc. of the 17th British Machine Vision Conference (BMVC), Edinburgh, UK, September 2006

[15] D. G. Lowe, Object Recognition from Local Scale-Invariant Features, Proc. of the International Conference on Computer Vision (ICCV), Vol. 2, Sept. 1999, Kerkyra, Greece, pp. 1150–1157.

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

[17] K. Mikolajczyk, B. Leibe, and B. Schiele. Multiple Object Class Detection with a Generative Model. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Vol. 1, June 2006, pp. 26–36.

[18] A. Opelt, M. Fussenegger, A. Pinz, and P. Auer, Weak Hypotheses and Boosting for Generic Object Detection and Recognition, European Conference on Computer Vision (ECCV), 2004, pp. 71–84.

[19] M. Stommel, O. Herzog, SIFT-Based Object Recognition with Fast Alphabet Creation and Reduced Curse of Dimensionality, Image and Vision Computing New Zealand (IVCNZ), Nov. 23.-25., 2009, Wellington, New Zealand.

[20] M. Varma, D. Ray, Learning The Discriminative Power-Invariance Trade-Off, IEEE International Conference on Computer Vision (ICCV) 2007.

[21] M. Weber, M. Welling, and P. Perona, Unsupervised Learning of Models for Recognition, European Conference on Computer Vision (ECCV), 2000, pp. 628–641.

[22] L. Yang, R. Jin, R. Sukthankar, and F. Jurie, Unifying Discriminative Visual Codebook Generation with Classifier Training for Object Category Recognition, In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008.

[23] J. Zhang, M. Marshalek, S. Lazebnik, and C. Schmid, Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study, Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06), 17–22 June 2006.