# Comparing feature matching for object categorization in video surveillance

# Comparing Feature Matching for Object Categorization in Video Surveillance

Rob G.J. Wijnhoven[1,2] and Peter H.N. de With[2,3]

[1] ViNotion B.V., 5612 AZ Eindhoven, The Netherlands
[2] Eindhoven University of Technology, Eindhoven, The Netherlands
[3] CycloMedia Technology B.V., 4180 BB Waardenburg, The Netherlands

**Abstract.** In this paper we consider an object categorization system using local HMAX features. Two feature matching techniques are compared: the MAX technique, originally proposed in the HMAX framework, and the histogram technique originating from Bag-of-Words literature. We have found that each of these techniques have their own field of operation. The histogram technique clearly outperforms the MAX technique with 5–15% for small dictionaries up to 500–1,000 features, favoring this technique for embedded (surveillance) applications. Additionally, we have evaluated the influence of interest point operators in the system. A first experiment analyzes the effect of dictionary creation and has showed that random dictionaries outperform dictionaries created from Hessian-Laplace points. Secondly, the effect of operators in the dictionary matching stage has been evaluated. Processing all image points outperforms the point selection from the Hessian-Laplace operator.

**Keywords:** video surveillance, object categorization, classification, HMAX framework, histogram, bag-of-words, random, Hessian-Laplace.

## 1 Introduction

Analysis tools have become an indispensable part of a security system with surveillance cameras due to the amount of video data processed by a security operator. The analysis and understanding of scenes starts typically with motion analysis and tracking of objects of interest. A further step is to classify objects in a number of predetermined categories.

Various approaches have been evaluated for object classification. The *Bag-of-Words (BoW)* model was first adopted from text-recognition literature by Csurka *et al.* in [1] for object categorization and has become a popular method for object classification [2,3,4,5,6]. The feature vector stores a histogram containing the number of appearances for each visual feature in a visual dictionary. Riesenhuber and Poggio [7] have proposed a biologically plausible system for object categorization, called *HMAX*. Conceptually, this system works in a comparable way to the BoW model. However, instead of storing a histogram of occurrences of the dictionary features, the distance value of the best match is stored for each feature. Where BoW models typically consider image points selected by *Interest Point Operators (IPOs)*, the HMAX model considers all image

J. Blanc-Talon et al. (Eds.): ACIVS 2009, LNCS 5807, pp. 410–421, 2009.

points. For both the BoW and the HMAX model, the dimensionality of the final feature vector is equal to the number of visual words. Although both methods have been presented and analyzed separately, an absolute comparison for the same dataset has not been published. The purpose of our comparison is to identify the best technique and consider relevant system aspects.

In this paper, we study an object categorization system based on a visual dictionary of local features and compare two techniques for feature vector generation. We show that each technique has a preferred field of operation, depending on the dictionary size. Aiming at an embedded implementation with limited computation power, choosing the best technique of the two for the actual field of operation gives a performance gain of up to 15% for a similar dictionary size and computational complexity.

The remainder of this paper is as follows. Section 2 describes the categorization system and the two compared feature matching techniques. Section 3 presents results on the comparison of the MAX and histogram techniques for both the visual dictionary creation and dictionary matching. Conclusions and recommendations for future work are given in Section 4.

## 2 System Description

The categorization system consists of several steps which are depicted in Figure 1. During the training stage, images from the training set (1a in Figure 1) are processed by an *Interest Point Operator* (IPO) (Block 2) to obtain characteristic locations in the image. Typically, IPOs find points that correspond to corner-points or blob-like structures. Several operators have been compared and evaluated in [8,9]. Next, descriptions of the local regions around the interest points are generated in Block 3. These local image descriptions are called *features*. A dictionary is created by selecting appropriate features (Block 4) and storing them in the visual dictionary (Block 5). After creating the visual dictionary, it is matched with each training image (Block 6) to generate a *feature vector*. This matching stage is referred to as the *feature matching* stage. Finally, a *classifier* (Block 7) uses these vectors for the training/test images to learn/determine the true object class.

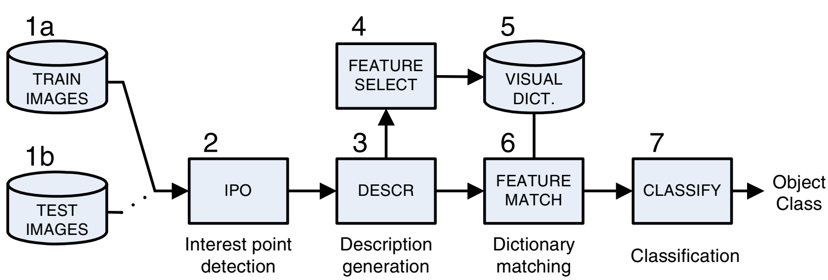



**Fig. 1.** System overview of the object categorization system

This paper concentrates particularly on the feature matching stage and evaluates two techniques for creation of the feature vector: the *MAX* and the *histogram* techniques. The MAX technique originates from the HMAX system, which is an overall framework for object categorization. Within this framework, we vary the feature matching technique applied and replace the original MAX technique with the histogram technique from the BoW models. Let us first describe the HMAX framework.

### 2.1 HMAX Framework

Since humans are good at object classification, it is reasonable to look into biological and neurological findings. Based on results from Hubel and Wiesel [10], Riesenhuber and Poggio have developed the "HMAX" model [11] that has been extended recently by Serre [12,13] and optimized by Mutch and Lowe [14]. We have implemented the model proposed by Serre up to the second processing layer [15]. The operation of the HMAX algorithm will now be addressed.

The algorithm is based on the concept of a feed-forward architecture, alternating between simple and complex layers, in line with the findings of Hubel and Wiesel [10]. The first layer implements simple edge detectors by filtering the gray-level input image with Gabor filters of several orientations and sizes to obtain rotation- and scale-invariance. The filters are normalized to have zero mean and a unity sum of squares. At each scale, the image is filtered in multiple orientations, resulting in so-called *S1 features*. For our experiments, we used the parameters as proposed by Serre *et al.* [13].

Continuing in the first layer, but as a succeeding step, the edge-filtered images are processed in the *C1 layer* to obtain invariance in local neighborhoods. This invariance will be created in both the spatial dimensions and in the dimension of scale. In order to obtain spatial invariance, the maximum is taken over a local spatial neighborhood around each pixel and the resulting image is sub-sampled. Because of the down-sampling, the number of resulting *C1 features* is much lower than the number of S1 features obtained in the preceding step. As an illustration, the resulting S1 and C1 feature maps for the input image of a bus at low-filtering scale are shown in Figure 2.

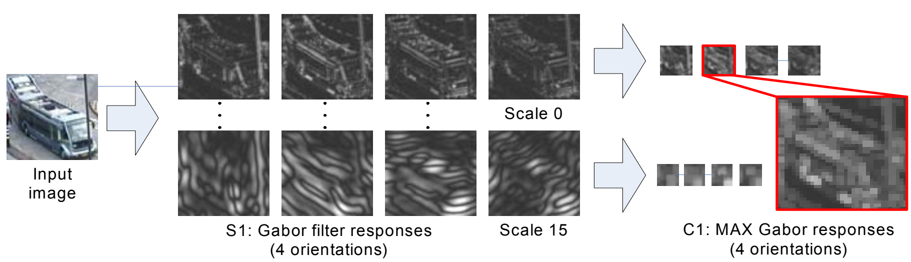



**Fig. 2.** HMAX feature space example (note: graylevel offset)

The second layer in the processing chain of the model matches stored dictionary C1 features with the C1 feature maps. The resulting matching scores are stored in the *S2 feature map*. The dictionary features are extracted from training images at a random scale and spatial location, at the C1 level. This random extraction is in accordance with Serre's [16] findings and has been confirmed by the authors [17]. Each feature contains all four orientations. Serre proposes to extract features at four different sizes: $4 \times 4$, $8 \times 8$, $12 \times 12$ and $16 \times 16$ elements. In our implementation, we use $5 \times 5$ features to enable a symmetric pixel neighborhood surrounding a central pixel and avoid large block sizes since in an evaluation they showed to be less important for categorization. Furthermore, for the computing of the subsequent *C2 layer*, for each dictionary feature, the match response with the smallest distance is extracted from the S2 feature map and stored in the final feature vector. This is done by taking the maximum S2 feature response over all scales and all spatial locations. Therefore, the final feature vector has a dimensionality equal to the number of dictionary features used.

The described HMAX framework is now linked to the system overview of Figure 1. In Block 2 involving interest point detection, all image positions at all scales are considered and referred to as the *AllPoints IPO*. In the description generation step (Block 3), the S1 and C1 feature maps are calculated. The dictionary matching stage (Block 6) computes the resulting S2 and C2 feature responses.

### 2.2 Bag-of-Words System: Histograms

Several systems for object recognition employ the *Bag-of-Words (BoW)* model using local features. Within this model, SIFT features [18] are broadly accepted [1,2,3]. The system is based on a dictionary of visual features (like the HMAX C1 features). The feature vector stores a histogram containing the number of appearances for each visual feature in the dictionary.

Conceptually, the HMAX system works in a comparable way to the BoW model. However, instead of storing a histogram of occurrences of the dictionary features in the BoW case, the best matching score for each feature is stored (*C2* value). The dimensionality of the final feature vector is, as in the BoW case, equal to the number of visual words.

As applied in literature [1,2,3,5,6,19], each considered position in the input image is compared to each dictionary feature and is *Vector Quantized (VQ)* to the most similar dictionary feature. The resulting feature vector stores the histogram value for each feature, representing the number of appearances for that feature, normalized to the total number of considered image points.

Because not every local image description is similar to a local feature in the visual dictionary, the vector quantization can result in a coarse quantization. This leads to noise in the feature vector, which is an inherently known degradation, as the local image description has a low matching score to every dictionary feature. Therefore, we propose a slightly different histogram technique. Instead of applying a *hard* quantization, we propose a more *soft* quantization, where increasing the histogram value of the most similar dictionary feature with unity is replaced by increasing the value by the corresponding matching score (distance). Therefore, the negative influence of image points that are not similar to any dictionary feature, is reduced. In the upcoming comparison, we refer to this technique as the *Matching Score (MS)*, in contrast to the original *Vector-Quantization (VQ)*.

### 2.3 Dictionary Creation

As proposed by Serre *et al.* [16], creation of the visual dictionary is best done by random sampling of features from images of natural scenery. Although this is counter intuitive, the authors have previously confirmed these results [17]. Applying interest point operators for dictionary creation is not useful within the HMAX framework. The difference in distinctiveness of the dictionary between dictionaries created from natural images, or images from the training set is however, insignificant. For the following experiments, we extract the visual dictionary from the training set. Although previous work [17] has shown that IPOs were not useful in the default HMAX framework, it is not clear if these findings hold when different feature matching techniques are applied. Furthermore, previously only dictionaries of 1,000 features were considered, while we enlarge the scale of operation to smaller and larger dictionaries.

### 2.4 Dictionary Matching

There are several ways to match the visual dictionary to an input image. In literature, typically interest point operators are used to select points that correspond to structures in the image (e.g. [2,3,4]). In contrast to considering the local image contents, random sampling can be applied, or all image points can be considered (grid-like sampling). It has been found that for dictionary matching, random and grid-like sampling can outperform interest point operators [4,5,6]. The original HMAX model applies a grid-like sampling, where all image points at all considered scales are matched with the dictionary (referred to as the *AllPoints* technique). The authors have previously compared several interest point operators for dictionary matching in the HMAX framework for a single dictionary size [17]. The computational complexity of the system (after dictionary creation) is linear to the number of considered image points. For embedded applications with limited computation power, methods that consider more image positions (like grid-like sampling) can therefore be inappropriate. Therefore, we investigate the effect on classification performance for visual dictionary matching on both the AllPoints and the Hessian-Laplace technique.

## 3 Experiments

First, we commence with defining the difference between training and testing and the performance measurement criterion. Given a set of *object classes* and an image containing one object, the task of object categorization is to determine the

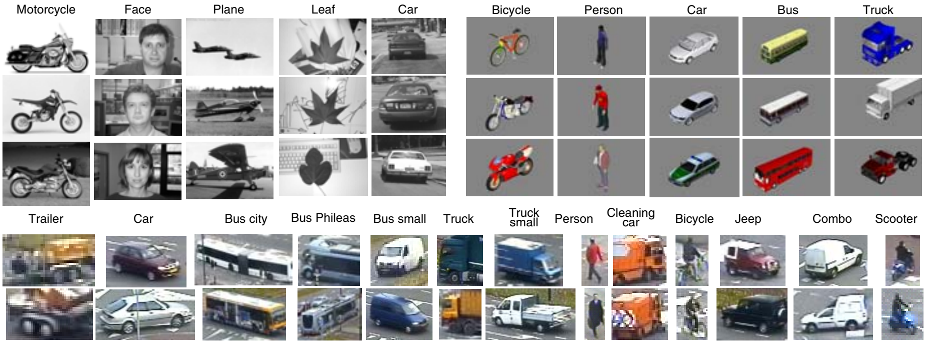



**Fig. 3.** Example images from datasets Caltech 5 (top-left), Wijnhoven2008 (top-right) and Wijnhoven2006 (bottom)

correct object-class label of the visualized object. The operation of object categorization systems is divided in two phases: training and testing. During training, the system learns from the *training set*, consisting of a number of example images for each object class. The performance of the algorithm is determined as the percentage of correctly labeled objects from the *test set*, averaged over all object classes.

We define the following datasets used for the evaluation. Three different categorization datasets are processed using the categorization system as presented in Section 2: a low-resolution dataset extracted from an hour of surveillance video (*Wijnhoven2006*, 13 classes), a synthetic traffic dataset (*Wijnhoven2008*, 5 classes), and the Caltech-5 dataset[1] (5 classes) containing faces, cars, leafs, planes and motorbikes. See Figure 3 for a visual overview.

Using these datasets, we create the visual dictionary in different ways and evaluate its influence on the performance of the MAX and histogram techniques. Next, the visual dictionary is matched to image points selected by different interest point operators and we measure the resulting performance of the same two feature matching techniques.

### 3.1 Dictionary Creation: Random vs. Hessian-Laplace

In this experiment we investigate two ways of creating the visual dictionary: random sampling and sampling around Hessian-Laplace interest points. In both cases, we create the initial large dictionary by sampling from images from the training set. To generate the final visual dictionary, a fixed number of features is randomly extracted from this initial set. Feature matching is applied with the techniques as discussed in Section 2: MAX and histogram. During dictionary matching, all image points are processed (*AllPoints* operator). A *Nearest Neighbor (NN)* classifier is used for the final classification.

[1] http://www.robots.ox.ac.uk/~vgg/data/data-cats.html

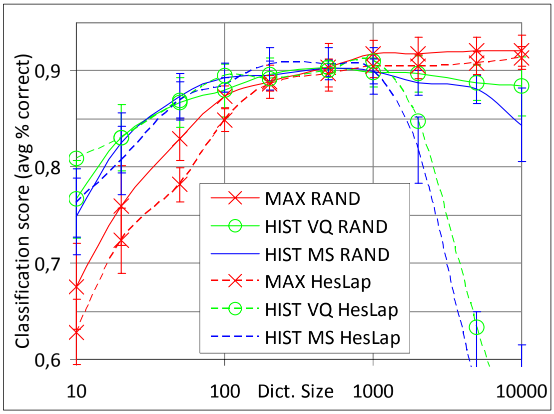
The results are visualized in Figure 4 and lead to several conclusions. First, we consider random dictionary generation (solid lines). It can be seen that for small dictionaries of up to 500 features, the histogram technique outperforms the MAX technique and obtains a gain of 5–15% in classification performance. For dictionaries larger than 500–1,000 features, the MAX technique is preferred. It is interesting to see that both techniques have their preferred field of operation and outperform each other. The computational complexity is equal for both the MAX and histogram techniques and is linear to the number of dictionary features. For computationally constrained systems, the histogram is clearly preferred, while for unconstrained systems, the MAX technique should be used.

Within the experiment, we have employed a vector quantization in the histogram creation procedure. To this end, we compare two cases: the *hard* Vector Quantization (VQ) and the *soft* Matching Score (MS) techniques, as discussed in Subsection 2.2. Figure 4 shows the results of these experiments. Overall, the VQ technique gives an improvement of a few percent in the classification score. Within the range of 50–500 features, there is no significant improvement, or even a small loss. For very small dictionaries of 10–20 features, the VQ technique gives a clear improvement. This is likely due to the large number of points assigned to only a small number of dictionary bins (features), so that the score per bin is always significant and the influence of noise is decreased.

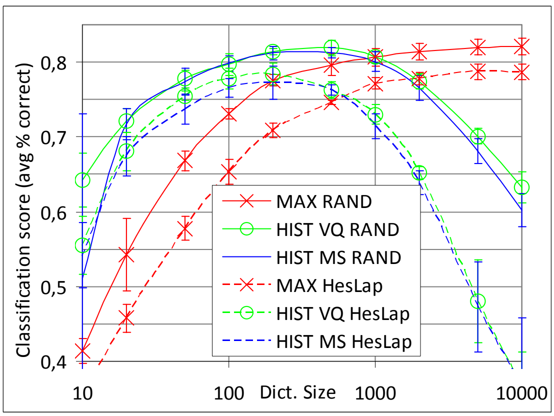
The differences in performance between the histogram and the MAX techniques can be explained by considering that the histogram technique stores the *distribution* of features over the image, whereas MAX only stores the *best response*. This makes the MAX technique very sensitive to variations in the maximum feature appearances. Moreover, when making histograms for large dictionaries, the number of dictionary features approaches the number of image positions, resulting in sparse, noisy histograms, which make the histogram approach less attractive.

Second, we compare dictionary generation using random selection and extraction around Hessian-Laplace interest points. Figure 4 shows that the results of the Hessian-Laplace technique (dashed lines) follow the results of the random technique (solid lines), with an overall lower performance of 5–10%. Towards large dictionaries, the performance of the Hessian-Laplace technique decreases drastically. Over the complete range of dictionary sizes, the random selection outperforms the Hessian-Laplace technique. A marginal exception are the very small dictionaries with 10–20 features, where the Hessian-Laplace slightly outperforms random selection. The conclusion holds for both the MAX technique and the histogram techniques. Thus, random dictionary creation is preferred over using the Hessian-Laplace technique.

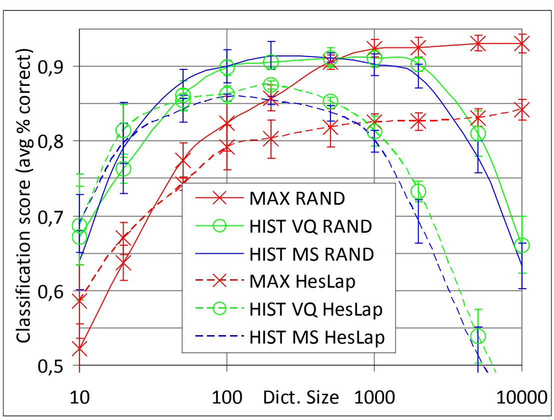
In previous work [17], the authors have already shown that for the HMAX framework, dictionary generation using random sampling outperforms the Hessian-Laplace operator. Previously, this conclusion was drawn for a fixed dictionary size. In the current experiments, we generalize this conclusion for a much larger field of operation. Our measurements show occasional exceptions for this conclusion when using very small dictionaries of less than 50 features.




(a) Caltech 5.

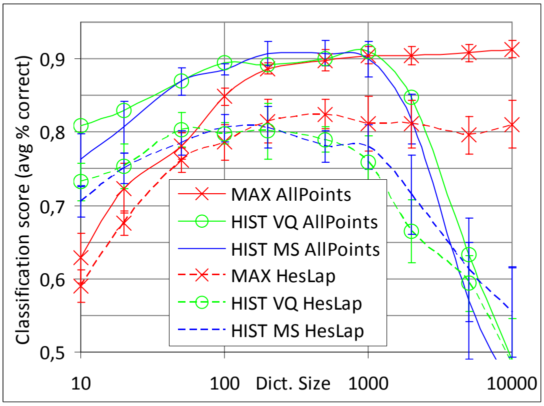



(b) Wijnhoven2006.

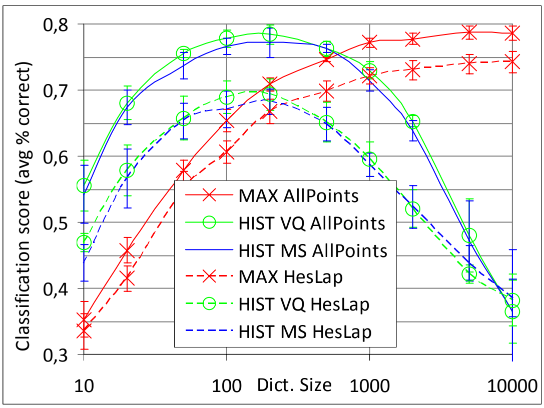



(c) Wijnhoven2008.

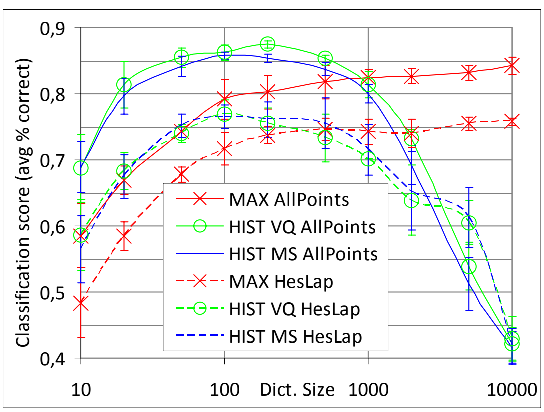
**Fig. 4.** Dictionary creation: Random and Hessian-Laplace (Matching: AllPoints)




(a) Caltech 5.




(b) Wijnhoven2006.




(c) Wijnhoven2008.

**Fig. 5.** Dictionary matching: AllPoints and Hessian-Laplace (Creation: Hessian-Laplace)

### 3.2 Dictionary Matching: AllPoints vs. Hessian-Laplace

In this experiment, as a preparation step, we first create dictionaries by HMAX features sampled around Hessian-Laplace interest points (It would have been more logical to exploit a random selection of features for the visual dictionary, but at the time of writing this article, those results were not yet available). Secondly, using these created dictionaries, we vary the interest point operator and measure the performance of dictionary matching. Both the *AllPoints* technique (considering all image points at all scales) and the Hessian-Laplace technique are evaluated for the interest point detection (Block 2 in Figure 1).

The results are shown in Figure 5. As can be seen, the AllPoints technique outperforms the Hessian-Laplace technique as an interest point operator in the dictionary matching procedure. The AllPoints performance is approximately 5–10% higher than the Hessian-Laplace performance. For the MAX technique, the authors have previously [17] shown that applying the Hessian-Laplace technique for dictionary matching results in lower performance. However, only a fixed size dictionary of 1,000 features was considered. The current results generalize the conclusions for the total dictionary size range.

For the histogram techniques, similar results can be seen: the performance of the Hessian-Laplace matching is generally lower that matching with all image points. Only for larger dictionaries, these conclusions are not valid. For very large dictionaries of 5,000 or more features, applying the Hessian-Laplace operator results in comparable or slightly higher classification performance.

In a secondary case of experiments, we have employed a vector quantization in the histogram creation procedure. To this end, we compare two cases: the *hard* Vector Quantization (VQ) and the *soft* Matching Score (MS) techniques, as discussed in Subsection 2.2. The results of these experiments can be seen in Figure 5. For the AllPoints matching, the VQ technique gives an improvement of a few percent in the classification score, for which no direct explanation can be given at this moment. For the Hessian-Laplace matching, a similar gain occurs for small dictionaries, but at some point, the performance of the VQ technique is slightly less than the MS processing. This decrease is explained by the creation of a more sparse histogram because Hessian-Laplace results in less image points than the AllPoints method. The authors expect that the quantization increases the noise in an already sparse distribution, leading to a performance decrease.

## 4 Conclusions

In this paper, we have addressed several aspects of an object categorization system using local HMAX features. Two feature matching techniques have been compared: the MAX technique, originally proposed in the HMAX framework, and the histogram technique originating from Bag-of-Words literature. The applied matching techniques are used for feature vector creation.

In the first experiment, two different ways of generating the visual dictionary were evaluated: extracting features at random and extracting around Hessian-Laplace interest points. In the second experiment, the interest point operators were varied in the dictionary matching stage. The AllPoints and the Hessian-Laplace interest point operators have been evaluated. We have found that for all experiments, each of these techniques have their own field of operation. The histogram technique clearly outperforms the MAX technique with 5–15% for small dictionaries up to 500–1,000 features. For larger feature sets, the MAX technique takes over and has superior performance. The computational complexity of both the MAX and the histogram technique is linear to the number of dictionary features and the number of matched image points (interest points). Aiming at an embedded implementation (surveillance), the histogram technique is favored over the MAX technique.

For the histogram dictionary matching, we have compared both the often used *hard vector Quantization (VQ)* technique and the proposed *soft Matching Score (MS)* technique for the histogram creation. Overall, VQ tends to give a small improvement in classification score.

We have compared different techniques for dictionary generation. Random extraction is preferred over extraction around Hessian-Laplace interest points, which typically results in a decrease in classification performance of 5–10%. These results are in line with earlier work of the authors [17] and is generalized in this paper to a large range of dictionary sizes.

Furthermore, the second experiment (comparing AllPoints and Hessian-Laplace for dictionary matching) shows that matching with the AllPoints operator outperforms the Hessian-Laplace interest point operator with 5–10%. This is in line with earlier findings [4,5,6]. This conclusion is a generalization of earlier work of the authors [17] which has been expanded here to a large range of dictionary sizes.

In the current experiments, the dictionaries were created using random selection from the initially large set that was constituted by random sampling or extraction around Hessian-Laplace points. Feature selection methods can be used that result in more distinctive visual dictionaries. Recent work of the authors [20] shows that this can result in a significant boost in classification performance.

## References

1. Csurka, G., Dance, C.R., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: Proc. European Conference on Computer Vision (ECCV) (May 2004)
2. Sivic, J., Russell, B.C., Efros, A.A., Zisserman, A., Freeman, W.T.: Discovering objects and their location in images. In: Proc. IEEE Int. Conf. on Computer Vision (ICCV), October 2005, vol. 1, pp. 370–377 (2005)
3. Sudderth, E.B., Torralba, A., Freeman, W.T., Willsky, A.S.: Learning hierarchical models of scenes, objects, and parts. In: Proc. IEEE Int. Conf. on Computer Vision (ICCV), October 2005, vol. 2, pp. 1331–1338 (2005)
4. Fei-Fei, L., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 524–531. IEEE Computer Society, Washington (2005)
5. Jurie, F., Triggs, B.: Creating efficient codebooks for visual recognition. In: Proc. IEEE Int. Conf. on Computer Vision (ICCV), October 2005, pp. 604–610 (2005)

6. Nowak, E., Jurie, F., Triggs, B.: Sampling strategies for bag-of-features image classification. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 490–503. Springer, Heidelberg (2006)
7. Riesenhuber, M., Poggio, T.: Hierarchical models of object recognition in cortex. Nature Neuroscience 2(11), 1019–1025 (1999)
8. Schmid, C., Mohr, R., Bauckhage, C.: Evaluation of interest point detectors. International Journal of Computer Vision 37(2), 151–172 (2000)
9. Mikolajzyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Van Gool, L.: A comparison of affine region detectors. Int. Journal on Computer Vision (IJCV) 65(1), 43–72 (2005)
10. Ullman, S., Vidal-Naquet, M., Sali, E.: Visual features of intermediate complexity and their use in classification. Nature Neuroscience 5, 682–687 (2002)
11. Riesenhuber, M., Poggio, T.: Models of object recognition. Nature Neuroscience 3, 1199–1204 (2000)
12. Serre, T., Wolf, L., Poggio, T.: Object recognition with features inspired by visual cortex. In: Proc. of Computer Vision and Pattern Recognition (CVPR), June 2005, pp. 994–1000 (2005)
13. Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., Poggio, T.: Robust object recognition with cortex-like mechanisms. Trans. Pattern Analysis and Machine Intelligence (PAMI) 29(3), 411–426 (2007)
14. Mutch, J., Lowe, D.G.: Multiclass object recognition with sparse, localized features. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), June 2006, vol. 1, pp. 11–18 (2006)
15. Wijnhoven, R., de With, P.H.N.: Patch-based experiments with object classification in video surveillance. In: Blanc-Talon, J., Philips, W., Popescu, D., Scheunders, P. (eds.) ACIVS 2007. LNCS, vol. 4678, pp. 285–296. Springer, Heidelberg (2007)
16. Serre, T.: Learning a Dictionary of Shape-Components in Visual Cortex: Comparison with Neurons, Humans and Machines, Ph.D. thesis, Massachusetts Institute of Technology Computer Science and Artificial Intelligence Laboratory (April 2006)
17. Wijnhoven, R., de With, P.H.N., Creusen, I.: Efficient template generation for object classification in video surveillance. In: Proc. of 29th Symposium on Information Theory in the Benelux, May 2008, pp. 255–262 (2008)
18. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. Journal of Computer Vision (IJCV) 60(2) (January 2004)
19. Crandall, D.J., Huttenlocher, D.P.: Weakly supervised learning of part-based spatial models for visual object recognition. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 16–29. Springer, Heidelberg (2006)
20. Creusen, I., Wijnhoven, R., de With, P.H.N.: Applying feature selection techniques for visual dictionary creation in object classification. In: Proc. Int. Conf. on Image Processing, Computer Vision and Pattern Recognition (IPCV) (July 2009)