A Hybrid Supervised-Unsupervised Vocabulary Generation Algorithm for Visual Concept Recognition

Alexander Binder[1], Wojciech Wojcikiewicz[1,2], Christina Müller[1,2], and Motoaki Kawanabe[1,2]

[1] Berlin Institute of Technology, Machine Learning Group, Franklinstr. 28/29, 10587 Berlin, Germany
{alexander.binder@, wojwoj@mail.}tu-berlin.de
[2] Fraunhofer Institute FIRST, Kekuléstr. 7, 12489 Berlin, Germany
{motoaki.kawanabe,christina.mueller}@first.fraunhofer.de

Abstract. Vocabulary generation is the essential step in the bag-of-words image representation for visual concept recognition, because its quality affects classification performance substantially. In this paper, we propose a hybrid method for visual word generation which combines unsupervised density-based clustering with the discriminative power of fast support vector machines. We aim at three goals: breaking the vocabulary generation algorithm up into two sections, with one highly parallelizable part, reducing computation times for bag of words features and keeping concept recognition performance at levels comparable to vanilla k-means clustering. On the two recent data sets Pascal VOC2009 and ImageCLEF2010 PhotoAnnotation, our proposed method either outperforms various baseline algorithms for visual word generation with almost same computation time or reduces training/test time with on par classification performance.

1 Introduction

Bag of words features [1] have turned into a widely-acknowledged tool for concept recognition which has shown superior performance in many recent contests on wide-domain image collections with high background and concept variance as well as presence of clutter [2–5]. Most prominent methods for visual vocabulary generation are unsupervised techniques like the density-based k-means algorithm, radius based clustering [6], and supervised methods like extremely randomized clustering forests (ERCF) [7–9]. Since people have tried more and more difficult images recently, one can observe an increase in the typical word size. See for example the 300 words used in [10] on the seminal Caltech101 benchmark [11] which has less clutter and rather low variance versus 4000 words [12] on Pascal VOC challenge data. Such an increase implies higher running times during visual vocabulary creation and bag of word computation. Several schemes have been proposed to deal with the runtime issue like hierarchical clustering [13, 14] or ERCF. From our own experience both methods can suffer

drawbacks in concept recognition performance compared to k-means clustering even though hierarchical k-means (HKM) speeds up notably over k-means and ERCF shows superb computational efficiency. In this paper, we propose a novel algorithm which uses the hierarchical clustering idea together with linear support vector machines (SVM) trained locally within each cluster and and has faster computation speeds in theory compared to vanilla k-means-based bag of words representations, while still maintaining the recognition performance of k-means visual vocabularies.

This paper is organized as follows. In Section 2, we explain our hybrid combination approach. After describing the datasets in Section 3 and the experimental setup in Section 4, we compare our method in Section 5 against k-means, hierarchical k-means and ERCF baselines on the two recent datasets Pascal VOC2009 and ImageCLEF2010 PhotoAnnotation.

2 Visual Word Generation

Bag of word features are based on three steps. At first one computes for each image a set of base features. In the second step the base features from the training data are used for computing a discretization of the input space of the base features into N regions. In the third step the base features extracted from one image are used to compute a histogram of dimensionality N based on assignments of the base features to the bins of the discretization obtained in the second step.

2.1 Hybrid Supervised-Unsupervised Approach

In order to generate a vocabulary of N visual words, we start with an unsupervised clustering of the base features into $N/2$ centers. For simplicity we relied on k-means clustering. At each of the clusters we train one support vector machine (SVM) in order to divide the cluster region into two parts. This gives rise to a partition of the space of the base features into N regions. The binary labels of the base features used for the SVM training are constructed from the image labelling which is inherited down to the base features belonging to one image, as we will explain in the following.

For multi-label concept recognition problems one issue remains to be solved. Each image can belong to several concept classes. At each cluster we have to select one partition of the set of all concept labels into two sets used for labelling the base features for binary SVM training. Since we are not interested in a perfect classification at a local cluster, we adopted an approximate randomized process to obtain good candidates for partitioning of the label set to be used to define a binary labeling.

The candidate labelling generation process was motivated by two ideas. We want to select a binarization such that

1. **balancedness:** the number of base features having a label in the positive class is approximately half of all base features within one cluster.

2. **overlap:** the number of base features which have at the same time labels in the positive as well as the negative class is low.

The second constraint comes from the fact that a base feature is assigned for SVM training to the positive class if it has at least one image label in the set of positive classes. We assign all image labels to the base features independent of the position of the base features within an image. This can lead to an adversary situation where a base feature is assigned to the positive class although its position in the image belongs to an object from the negative class. Such a problem can be avoided in an object detection scenario with bounding box or object position information which is not available here.

We implemented the first constraint as follows. At first, starting from an empty set for positive classes, we randomly draw a class from the set of all concepts and add to the positive set. Then, we iterate this procedure, i.e. we add a class selected randomly from the set of the remaining categories. This gives a series of growing sets of positive classes. For each of these sets within the series we can count how many of the base features will be labeled positive, because they have at least one label belonging to the set of positive classes. Let S be the set of all base features and $l(S)$ be its original multi-label vector, then we count

$$bal(positives) := | \, |S|/2 - |\{s \in S \mid l(s) \cap positives \neq \emptyset\}| \, |, \tag{1}$$

where $|S|$ denotes the cardinality of the set S, i.e. the number of its elements. We select the set from this series which has the number of base features without the labels in the positive sets being closest to half of the total number of base features.

This procedure can be repeated M times to obtain M candidates which get subsequently checked for their overlap constraint. For the overlap constraint we count directly

$$ol := \{s \in S \mid l(s) \cap positives \neq \emptyset, \, l(s) \cap negatives \neq \emptyset\}, \tag{2}$$

which is the number of the base features labeled with concepts in positive and negative sets simultaneously and select the best T candidates to be used to define a binary labelling for SVM training.

Each of the T candidates was evaluated using five-fold cross validation in order to select one final classifier used at a local cluster node. This is admittably inspired by the ERCF algorithm which also uses a randomized generation of candidate partitions. On the other hand, ERCF chooses local dichotomies along one axis and deploys an entropy criterion. The pseudo-code for the two-class labeling procedure is summarized at the next page.

2.2 Relation to the Baseline Procedures

Figure 1 illustrates the proposed method in comparison with the three baselines with a synthetic example of two class data marked with red and blue. The

Generation of Candidates for Two-Class Labeling for SVM Training

```
choose M = number of random trials,
       T = number of candidates for SVM training
input: S = set of base features, l(S) their multi-labels
input: num_concepts= number of concepts in multi-label problem
output: Candidates = T partitions of the set of all concepts into two

for m=1:M
  positives(m,0) = {}

  for index=1:num_concepts
    class = random_select( concepts\positives(m,index-1) )
    positives(m,index) = {positives(m,index-1),class}
  end for

  i = argmin_{index} bal( positives(m,index) )
  mid(m) = positives(m,i)
  compute ol( mid(m) )
end for
Candidates = the T elements from mid with smallest overlap ol
```

proposed procedure is comparable to hierarchical k-means (HKM) with $N/2$ clusters at the top level and 2 clusters at the second level. Because we deploy a supervised technique instead of k-means with 2 centers at each cluster, the proposed method can capture the class information correctly. On the other hand, k-means and HKM fail to separate the red and the blue classes. ERCF uses image labels as well, however its appealing speed gains come from restriction of the partition process to axis-parallel splits of the input space. The class boundary in Figure 1 is however not aligned to the axes, thus the proposed procedure works best in Figure 1.

Compared to vanilla k-means we have practical speed-ups of visual vocabulary generation with the proposed method, because our method requires a smaller number of clusters in the initial step. The local classifier training step consists of $N/2$ independent jobs which can be run separately on a vanilla CPU cluster, thus computation time of the extra step is negligible.

Another advantage comes at the step of computing the bag of words histograms: k-means requires for each base feature to compute N distances. This can however be speeded up empirically by computing a distance matrix between cluster centers and employing the triangle inequality to exclude certain candidate centers. The proposed method uses $N/2$ distance computations and one SVM function evaluation. For the linear SVMs we used here this amounts to computing one inner product. Note that for normalized base features $\|f\|^2 = const$ the distance computation is equivalent to an inner product

$$\|f - g\|^2 = 2c - 2\langle f, g\rangle \tag{3}$$

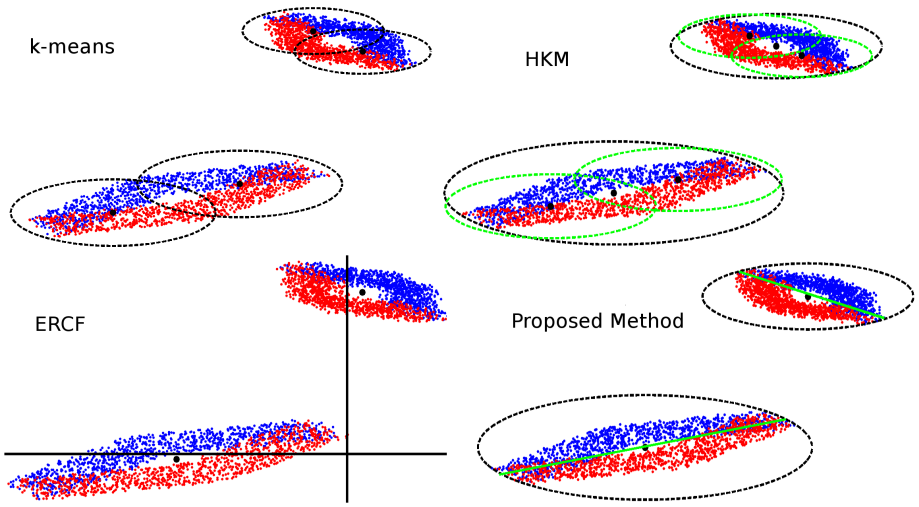

Fig. 1. An example of two blobs of two class data, marked red and blue and the results of different clustering algorithms with 4 clusters: Upper Left: k-means, Upper Right: Hierarchical k-means, Lower Left: ERCF, Lower Right: The proposed method.

Thus, our method requires a computational amount of $N/2 + 1$ inner products compared to N inner products for vanilla k-means.

The advantage in computational speed enables us to double the word sizes of the standard k-means vocabularies with tiny extra runtime. Although the proposed method requires the same amount of time for feature computation as a k-means-based visual vocabulary, it increases the time to constructing kernels by a factor of two. We remark that this is still acceptable, because vocabulary generation is the bottle-neck in the entire process.

We have pursued a hybrid algorithm for visual word generation, where an unsupervised clustering method is done prior to the local supervised classification. This is because we do not expect to find a reasonable linear separation on the global input set of all base features. We conjectured that pure supervised procedures on entire base features such as ERCF can potentially suffer from degraded performance due to this problem. Furthermore, they also may have computational difficulties. The large cardinality in the order of hundred thousands or millions of input features necessary for visual word generation algorithms slows down linear SVMs and is still prohibitive for non-linear SVMs.

In this paper, we presented one special instance of an interpolation between unsupervised clustering and local classification. In general one can generate N visual words by using unsupervised clustering to obtain $N \cdot 2^{-k}$ base clusters and train k supervised classifiers at each cell which generates 2^k additional words at the $N \cdot 2^{-k}$ clusters.

Fig. 2. Pascal VOC2009 example images.

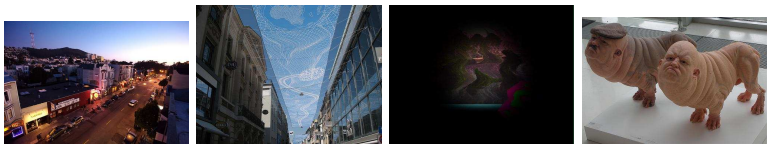

Fig. 3. ImageCLEF2010 example images.

3 Datasets

Pascal VOC2009 data set has been used for the Pascal Visual Object Classification Challenge [4]. The part with disclosed labels is comprised of 7054 images falling into twenty object classes. The objects typically have highly varying sizes, positions and backgrounds.

ImageCLEF2010 PhotoAnnotation data set has in its labeled part 8000 images from flickr with 93 concept classes with highly variable concepts containing well defined objects as well as many rather ambiguously defined concepts like Citylife, Cute or Visual_Arts which makes it highly challenging for any recognition system.

4 Experimental Setup

As the base features, we computed for each image grey and rgb SIFT features [15] on a dense grid of pitch size six. Our choice is a reduced set inspired by the winners' procedures of ImageCLEF2009 PhotoAnnotation and Pascal VOC2009 [4]. The base features are clustered with the proposed method, and three baselines, i.e. vanilla k-means, hierarchical k-means (HKM) and ERCF. HKM extracts $N/2$ clusters at the top and 2 clusters at the lowest level which is close to the proposed method. Due to huge number of all base features, we used randomly drawn SIFT features for clustering: 2.4 millions for the grey color channel and 800000 for rgb-SIFT which has the triple dimensionality of grey SIFT. We have chosen these two color channels exemplarily for their different dimensionality as it is known that the quality of density estimation which is implicitly performed by k-means may deteriorate in higher dimensions.

The local SVMs were trained with five-fold cross validation over regularization constants 0.25, 1 and 4. For each cluster, the best SVM was selected from 10 candidates generated by our procedure based on the cross validation scores. Due to computational costs, we limited for SVM training the number of base features to 5000 by random selection from all base features in each local cluster.

For visual concept classification, we deployed the χ^2-kernel based on bag-of-word features whose width is set to the average χ^2-distance. Then, all kernels were normalized to standard deviation one in Hilbert space, which allows to use the regularization constant 1 as a good approximative rule of thumb for SVM training, where we employ the shogun toolbox [16]. The performance is measured with average precision (AP) and area under curve (AUC) which are both threshold-invariant ranking measures. We evaluate all settings with 10 random splits to see statistical significance.

In order to advantages of proposed method in performance and runtime, we considered the following two settings.

Experiment 1. Comparison between the vocabularies with the same size of 500 words.

Experiment 2. Comparison of vanilla k-means with 4000 words and the other vocabularies with 8000 words.

In the first case, the vocabularies except for k-means can be computed much faster. Therefore, it is still OK, if the alternatives perform at least on par with k-means. In the second case, the larger 8000 vocabularies can be generated easily from the 4000 k-means prototypes by our algorithm and HKM with small computational costs, while ERCF can construct 8000 words quickly. Here, we are interested in performance gains over the standard bag-of-words procedure based on k-means.

5 Results

5.1 Concept Recognition Performance

The recognition performances using rgb SIFTs are summarized in Table 1 and 2 for VOC2009, and 3 and 4 for ImageCLEF2010. Results using grey SIFTs with qualitatively the same outcome are given in Appendix.

In Experiment 1, we compared vocabularies with 500 word in total. For both data sets (Table 1 & 3, the proposed method achieved slightly higher scores than k-means within shorter runtime, while the other faster variants degraded their performances to some extent. We further tried variants of our method with early-stopped clusters, where we reduced the number of k-means iterations to five. The performance did not drop much, because the clustering served only as a rough initialization for local classifications. This suggests that an exact density based clustering does not play a too large role and sheds an interesting light on claims that density-based clustering is inferior to alternatives such as radius-based clustering [17, 6].

In Experiment 2, we compared vocabularies whose sizes are closer to the ones used in recent competitions (k-means with 4000 words and the faster methods with 8000 words). Note that we did not compute vanilla k-means 8000 as clustering would take almost two weeks and is deemed too slow given the used setting. This is another way of fair comparisons, i.e. under equal time limitations. For VOC2009, our approach achieved notable gain over the k-means baseline, while HKM improved only slightly or ERCF even lost against the baseline. On the other hand, on ImageCLEF2010, all larger vocabularies of size 8000 did not improve the 4000 k-means significantly. We will see that for some of the abstract concepts in ImageCLEF2010 our algorithm degraded recognition performances.

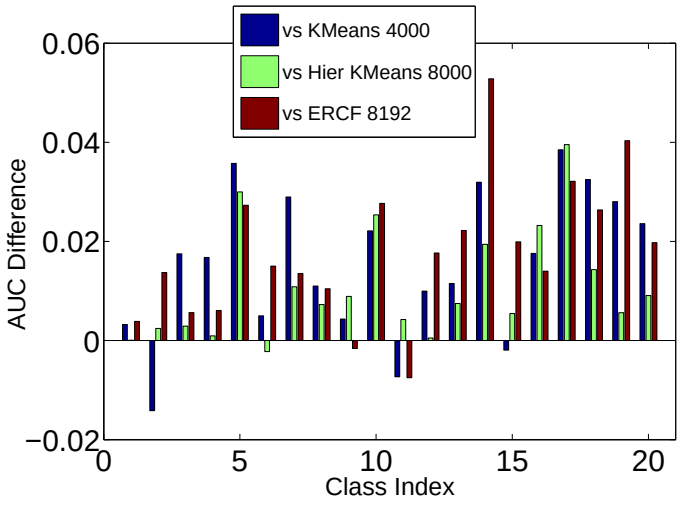

Fig. 4. Class-wise differences by AUC for VOC2009, rgb channel between proposed method, 8000 words and vanilla k-means 4000 words (left), hierarchical k-means 8000 words (mid), ERCF 8192 words (right).

Table 1. Recognition performances of the baselines with 500 words versus those by our approach with 250 clusters and local SVMs (Experiment 1) on VOC2009 (summary from 10 repetitions).

Method / Score	AP	AUC
baseline: rgb-SIFT, hierarch KM250x2	43.87 ± 5.15	85.83 ± 1.71
baseline: rgb-SIFT, ERCF4x128	42.19 ± 5.33	85.39 ± 1.53
baseline: rgb-SIFT, KM500	44.46 ± 5.58	86.14 ± 1.78
proposed: rgb-SIFT, KM250, 5 iters+250 lin SVM	**45.16** ± 5.28	86.19 ± 1.70
proposed: rgb-SIFT, KM250+250 lin SVM	44.99 ± 5.25	**86.50** ± 1.45

Table 2. Recognition performances of the baselines with 4000/8000 words versus those by our approach with 4000 clusters and local SVMs (Experiment 2) on VOC2009 (summary from 10 repetitions).

Method / Score	AP	AUC
baseline: rgb-SIFT, hierarch KM4000x2	50.04 ± 5.18	88.04 ± 1.62
baseline: rgb-SIFT, ERCF16x512	47.54 ± 5.11	87.32 ± 1.54
baseline: rgb-SIFT, KM4000	48.94 ± 5.08	87.54 ± 1.73
proposed: rgb-SIFT, KM4000+4000 lin SVM	**52.70** ± 5.41	**89.11** ± 1.64

By inspecting the differences between the proposed method and vanilla k-means in Figure 4 (left) and 5, we see some gains on most classes and a small fraction of setbacks For VOC2009 data we observe larger gains for classes bottle(5), cow(10) and pottedplant(16) which belonged to the rather difficult classes according to their performance on test data results for the winners' submission. For ImageCLEF2010 data the trends are more diverse. For the rgb channel we lose performance with the proposed method in 15 concepts out of 93 like birthday(65),grafitti(67),abstract(72),cat(76) and bicycle(82), while having many gains across a variety of narrow and broad concepts like in Partylife(1), River(14), Motion_Blur(39), Architecture(53), Visual_Arts(66), Train(84), Skateboard(86) and Child(90). We assume that it is harder to create a meaningful binarization of all concepts into two classes when the number of concepts increases from 20 in VOC2009 to 93 in ImageCLEF2010. Here using a multi-class classification with several classes of approximately equal size could turn out to be beneficial.

5.2 Runtime Considerations

For 8000 visual words the classifier training required merely about 10 minutes on a 128 CPU cluster when using liblinear [18]. The linear SVMs are comparably fast despite the involved optimization problem, because they run on a

Table 3. Recognition performances of the baselines with 500 words versus those by our approach with 250 clusters and local SVMs (Experiment 1) on ImageCLEF2010 (summary from 10 repetitions).

Method / Score	AP	AUC
baseline: rgb-SIFT, hierarch KM250x2	32.53 ± 0.86	73.55 ± 1.30
baseline: rgb-SIFT, ERCF4x128	32.50 ± 1.40	73.62 ± 1.56
baseline: rgb-SIFT, KM500	33.07 ± 1.03	73.91 ± 1.44
proposed: rgb-SIFT, KM250, 5 iters+250 lin SVM	33.45 ± 0.94	74.38 ± 1.51
proposed: rgb-SIFT, KM250+250 lin SVM	**33.58** ± 0.92	**74.40** ± 1.41

Table 4. Recognition performances of the baselines with 4000/8000 words versus those by our approach with 4000 clusters and local SVMs (Experiment 2) on ImageCLEF2010 (summary from 10 repetitions).

Method / Score	AP	AUC
baseline: rgb-SIFT, hierarch KM4000x2,	36.29 ± 1.28	76.20 ± 1.50
baseline: rgb-SIFT, ERCF16x512	36.48 ± 1.19	76.04 ± 1.48
baseline: rgb-SIFT, KM4000,	36.16 ± 1.18	75.97 ± 1.40
proposed: rgb-SIFT, KM4000+4000 lin SVM	**36.78** ± 1.19	**76.60** ± 1.44

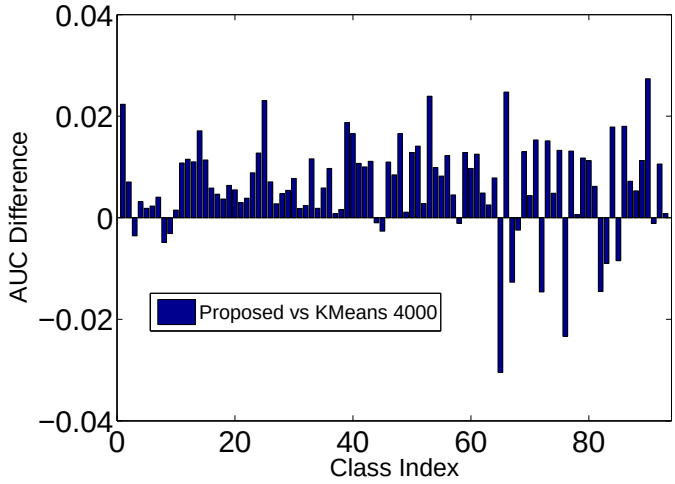

Fig. 5. Class-wise differences by AUC for CLEF2010, rgb channel between proposed method, 8000 words and vanilla k-means, 4000 words (other 2 omitted for readability).

small local set of features only. The k-means distance computation step is a simpler algorithm but gets executed globally on the set of all features (800000 for rgb SIFT, 2.4 Mio for grey SIFT). A single core k-means implementation required two hours for each iteration of k-means in order to generate 4000 visual words. For 8000 words the time would roughly be doubled. Typically multi-core implementations of k-means can be used to speed up this process as well as the unsupervised part in our proposed method, however they tend to end at parallelizations to 8 CPUs or require specialized hardware to use more cores. The proposed approach allows the distribution of the supervised classification part to independent CPUs. It has been already mentioned in Section 2 that our algorithm requires during bag of word computation time for N visual words $N/2 + 1$ inner product evaluations compared N such steps for vanilla k-means. This theoretical claim is consistent with the average feature computation times we observed per image for rgb-SIFT: KM4000 takes 58.73 s, ERCF16x512 takes

7.8 s, hierarchical KM4000x2 requires 64.09, a bag of words feature based on KM8000 based on a prematurely terminated clustering needs 128.43 s. The running times for bag of words computation using such kind of clustering are fully comparable, whereas classification performance might be slightly degraded. The proposed method KM4000 + 4000 linear SVMs takes 68.28 s which is in the range of the other k-means based methods which include a vocabulary of size 4000.

In practice running times are affected by many additional factors. Our system for example stores sift features in bzip2-ed form for keeping disk space within reasonable bounds. This requires to uncompress them in memory which adds a certain offset to each bag of words computation step.

5.3 Label Distribution Statistics across Visual Words

We have seen in Section 5.1 that the proposed method performs better in terms of error measures. The performance was examined class-wise in the aforementioned section. Now we would like to assume the word-wise viewpoint. For sanity checking we examine whether the usage of local classifiers leads to a difference in base feature assignments to visual words. To this end, we computed for each visual word the entropy of label distribution given by all base features which are assigned to the visual word in question. Consider a visual vocabulary V and a set of K base features with $\{0,1\}$-valued multi-labels from C concepts $\{(b_i, y_{ji}), j = 1, \ldots, C, i = 1, \ldots, K\}$. Define using proper normalization

$$p(y_c, v) \propto \sum_{i|v=\arg\min_{w \in V} d(b_i,w)} y_{ci} \tag{4}$$

We can compute the corresponding entropy $\tau(v)$ of the label distribution for a fixed word [19].

$$\tau(v) = -\sum_{j=1}^{C} p(y_j|v) \log(p(y_j|v)) \tag{5}$$

A lower entropy suggests a better separation of base features assigned to different labels for a given word. To compare a visual vocabulary as a whole we consider the cumulative sums of distribution of these entropies over the set of all visual words in a vocabulary. Figure 6 compares this distribution between the proposed method and hierarchical k-means with 4000x2 clusters, which is structurally the closest baseline, both using 8000 words. We observe that the proposed method has a higher mass of visual words at lower values of the label distribution entropy, which indicates that more informative words about some visual concepts were selected.

6 Conclusion

In this paper, we proposed a hybrid algorithm of unsupervised clustering and supervised linear SVMs for visual codebook generation. On VOC2009 and Im-

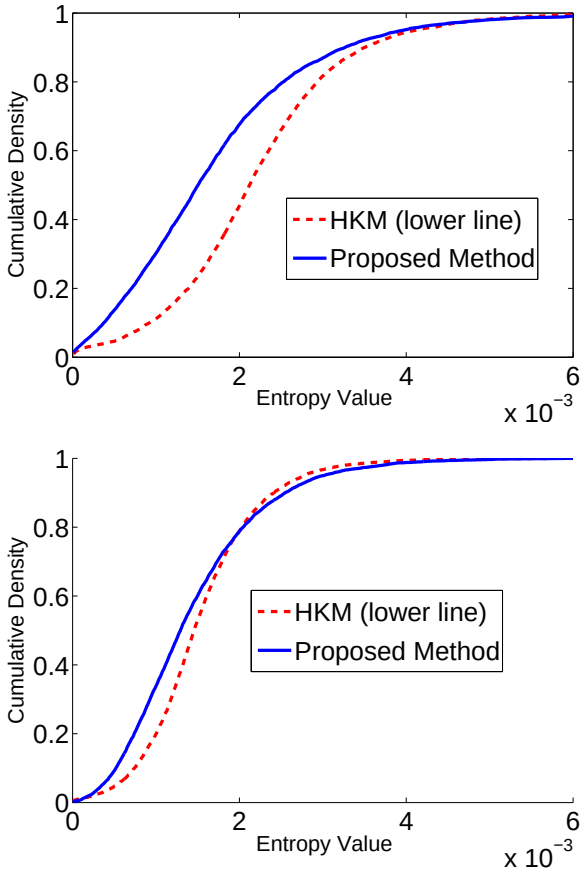

Fig. 6. Cumulative distribution functions of the label entropy across all visual words, rgb channel for the proposed method, 8000 words and hierarchical k-means, 8000 words, Upper: CLEF2010, Lower: VOC2009.

ageCLEF data sets, we showed clear advantages either in recognition performance or in computation speed over purely unsupervised (e.g. k-means, HKM) and purely supervised (e.g. ERCF) procedures. In particular, our approach can reduce runtime of word generation and assignment almost half without losing performance, while the fastest choice ERCF often loses unignorable amounts in performance scores. Furthermore, we can double the standard k-means vocabularies with small extra costs, which brought substantial improvements in VOC2009 and for majority of concepts in ImageCLEF2010. Further research should be done to incorporate fast linear multi-class classifiers or to find better binarization procedures used for the SVM training.

Acknowledgement. This work was supported in part by the German Federal Ministry of Economics and Technology (BMWi) under the project THESEUS (01MQ07018).

References

1. Csurka, G., Bray, C., Dance, C., Fan, L.: Visual categorization with bags of keypoints. In: Workshop on Statistical Learning in Computer Vision, ECCV, Prague, Czech Republic (2004) 1–22
2. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2007 (VOC2007). http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html (2007)
3. Tahir, M., van de Sande, K., Uijlings, J., Yan, F., Li, X., Mikolajczyk, K., Kittler, J., Gevers, T., Smeulders, A.: Surreyuva srkda method. http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2008/workshop/tahir.pdf (2008)
4. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2009 (VOC2009). http://www.pascal-network.org/challenges/VOC/voc2009/workshop/index.html (2009)
5. Nowak, S., Dunker, P.: Overview of the CLEF 2009 large-scale visual concept detection and annotation task. In: Working Notes of CLEF 2009 Workshop. (2009)
6. Jurie, F., Triggs, B.: Creating efficient codebooks for visual recognition. In: ICCV05. (2005) I: 604–610
7. Moosmann, F., Triggs, B., Jurie, F.: Fast discriminative visual codebooks using randomized clustering forests. In: Advances in Neural Information Processing Systems. (2006)
8. Moosmann, F., Nowak, E., Jurie, F.: Randomized clustering forests for image classification. IEEE Transactions on Pattern Analysis & Machine Intelligence **30** (2008) 1632–1646
9. Uijlings, J., Smeulders, A., Scha, R.: Real-time bag-of-words, approximately. In: CIVR. (2009)
10. Bosch, A., Zisserman, A., Muñoz, X.: Representing shape with a spatial pyramid kernel. In: Proceedings of the 6th ACM international conference on Image and video retrieval (CIVR '07). (2007) 401–408
11. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. In: Workshop on Generative-Model Based Vision. (2004)
12. van de Sande, K.E.A., Gevers, T., Snoek, C.G.M.: Evaluating color descriptors for object and scene recognition. IEEE Trans. Pat. Anal. & Mach. Intel. (2010)
13. Nister, D., Stewenius, H.: Scalable recognition with a vocabulary tree. In: CVPR '06: Proceedings of Conference on Computer Vision and Pattern Recognition. (2006)
14. Fulkerson, B., Vedaldi, A., Soatto, S.: Localizing objects with smart dictionaries. In: Proceedings of the European Conference on Computer Vision (ECCV). (2008)
15. Lowe, D.: Distinctive image features from scale invariant keypoints. International Journal of Computer Vision **60** (2004) 91–110
16. Sonnenburg, S., Rätsch, G., Henschel, S., Widmer, C., Behr, J., Zien, A., de Bona, F., Binder, A., Gehl, C., Franc, V.: The shogun machine learning toolbox. Journal of Machine Learning Research (2010)

17. van Gemert, J., Geusebroek, J., Veenman, C., Smeulders, A.: Kernel codebooks for scene categorization. In: ECCV. (2008)
18. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: Liblinear: A library for large linear classification. J. Mach. Learn. Res. **9** (2008) 1871–1874
19. Wojcikiewicz, W., Binder, A., Kawanabe, M.: Enhancing image classification with class-wise clustered vocabularies. In: Proceedings of the 20th International Conference on Pattern Recognition (ICPR). (2010)

7 Appendix

7.1 Results for Grey SIFT Features

Table 5. Results for VOC2009 dataset with grey SIFT , 10-fold cross-validation.

Method / Score	AP	AUC
baseline: SIFT, hierarch KM250x2 words	43.27 ± 5.23	85.48 ± 1.93
baseline: SIFT, ERCF4x128 words	41.11 ± 5.21	84.84 ± 1.54
baseline: SIFT, KM500	44.03 ± 5.74	85.81 ± 1.90
proposed: SIFT, KM250, 5 iter+250 lin SVM	44.28 ± 5.75	**85.94** ± 1.91
proposed: SIFT, KM250 +250 lin SVM	**44.60** ± 5.18	85.93 ± 1.86
baseline: SIFT, hierarch KM4000x2	49.49 ± 5.03	87.71 ± 1.67
baseline: SIFT, ERCF16x512	52.04 ± 5.14	88.60 ± 1.71
baseline: SIFT, KM4000	51.28 ± 5.59	88.47 ± 1.47
proposed: SIFT, KM4000+4000 lin SVM	**52.07** ± 5.08	**88.85** ± 1.63

Table 6. Results for CLEF2010 PhotoAnnotation dataset with grey SIFT, 10-fold cross-validation.

Method / Score	AP	AUC
baseline: SIFT, hierarch KM250x2	32.37 ± 1.04	73.49 ± 1.28
baseline: SIFT, ERCF4x128	31.48 ± 1.28	72.57 ± 1.84
baseline: SIFT, KM500 words	32.62 ± 0.96	73.57 ± 1.40
proposed: SIFT, KM250, 5 iters+250 lin SVM	32.31 ± 1.08	73.49 ± 1.41
proposed: SIFT, KM250, 120 iters+250 lin SVM	**32.78** ± 1.16	**73.82** ± 1.39
baseline: SIFT, hierarch KM4000x2	35.42 ± 1.09	75.78 ± 1.36
baseline: SIFT, ERCF16x512	34.82 ± 1.05	75.00 ± 1.50
baseline: SIFT, KM4000	35.13 ± 1.32	75.30 ± 1.32
proposed: SIFT, KM4000+4000 lin SVM	**35.74** ± 1.26	**76.04** ± 1.25