University of Marburg at TRECVID 2010: Semantic Indexing

Markus Mühling, Ralph Ewerth, and Bernd Freisleben

Department of Mathematics and Computer Science
University of Marburg, D-35032 Marburg, Germany
{muehling, ewerth, freisleb}@informatik.uni-marburg.de

Abstract

In this paper, we summarize our results for the semantic indexing task at TRECVID 2010. Last year, we showed that the use of object detection results as an additional input for SVM-based concept classifiers improved the overall performance.

This year, we investigated whether a state-of-the-art bag-of-visual-words (BoW) approach can also be improved by adding object-based features. In this context, Multiple Kernel Learning (MKL) was applied to find the best feature weighting.

The experiments revealed that the supplementation of BoW-based features with object-based features significantly improved the concept detection performance. Furthermore, we showed that a more uniform distribution of kernel weights using l_2-norm MKL gained better results.

Altogether, our best run achieved a mean inferred average precision of 6.96% and we submitted the best results for the concepts "vehicle" and "ground_vehicle".

1. Structured Abstract

The results of our participation in the semantic indexing task (also known as high-level feature extraction task) are presented in this section in form of the requested structured abstract. In the following sections, we describe our system for semantic indexing along with the experimental results. In Section 2, the different feature types are explained. The Multiple Kernel Learning framework is discussed in Section 3, while the experimental results are presented in Section 4. Section 5 concludes the paper.

"What approach or combination of approaches did you test in each of your submitted runs?"

The following four runs of category "A" were submitted:

- F_A_Marburg1_4: Baseline (RGB-SIFT)
- F_A_Marburg2_3: Baseline plus object-based features using l_1-norm MKL
- F_A_Marburg3_2: Baseline plus object-based features using l_2-norm MKL
- F_A_Marburg4_1: Baseline plus object-based features and global features

"What, if any significant differences (in terms of what measures) did you find among the runs?"
"Based on the results, can you estimate the relative contribution of each component of your system/approach to its effectiveness?"
We investigated the supplementation of state-of-the-art BoW-based features with object-based features. The BoW representation relies on RGB-SIFT descriptors using a dense sampling strategy. Different feature representations are combined using sparse (l_1-norm) and non-sparse (l_2-norm) MKL, respectively.
The runs considering object-based features clearly improved the overall performance. Our best run combining object-based and BoW-based feature representations using non-sparse MKL achieved a performance of 6.96% mean inferred average precision compared to 5.58% of the baseline system. Although several concepts of the category "scene" profited from additional global color and Gabor histograms, the overall performance was slightly decreased compared to the reference system.

"Overall, what did you learn about runs/approaches and the research question(s) that motivated them?"
The experiments revealed that the approaches exploiting object-based features significantly improved the overall performance compared to the baseline system. Some concepts like "animal", "bicycling" or "vehicle" were improved by more than 100%. For

"vehicle" and "ground_vehicle", we obtained the best results with 20.1% and 20.2%, respectively, in terms of inferred average precision among all submitted runs. Furthermore, the experiments showed that a more uniform distribution of kernel weights achieved better results than using l_1-norm MKL.

2. Feature Extraction

Based on the success of object-based features in our last year's system [12][13], we incorporated further specialized object detectors trained on separate public data sets. Since state-of-the-art semantic concept detection systems rely on the BoW approach, our current baseline system employs this feature representation. In Section 2.1, we present the BoW approach, followed by the object-based features in Section 2.2 and the global features in Section 2.3.

2.1 Bag-of-Visual-Words

We performed a dense sampling strategy to extract SIFT [10] descriptors at sampled keypoints, because the sparse representation using keypoint detectors like Harris-Laplace or DoG is often insufficient to describe natural images. To extract dense SIFT features, the Vision Lab Features Library (VLFEAT) [17] was used. It provides a fast algorithm for the calculation of a large number of SIFT descriptors of densely sampled features of the same scale and orientation. The SIFT descriptor geometry is specified by the number and size of the spatial bins and the number of orientation bins. A sampling step size of 5 pixels, 8 orientation bins and 4x4 spatial bins of sizes 4, 6 and 8 pixels were used. Thus, the resulting keypoint descriptors form a 128-dimensional feature vector.
Similar to the representation of documents in the field of text retrieval, an image can be represented as a bag of visual words that are quantized local image descriptors. The visual vocabulary is generated from a set of training images by clustering the extracted keypoint descriptors in their feature space and interpreting the cluster centers as visual words. Due to the huge amount of keypoints we only used 10 positively labeled training shots respectively keyframes per concept to construct a 1000-dimensional vocabulary using K-means. Based on this vocabulary, histograms were generated per shot by mapping the bag of descriptors from a keyframe to the visual words. Instead of just increasing the nearest neighbor, we used a soft-weighting scheme.

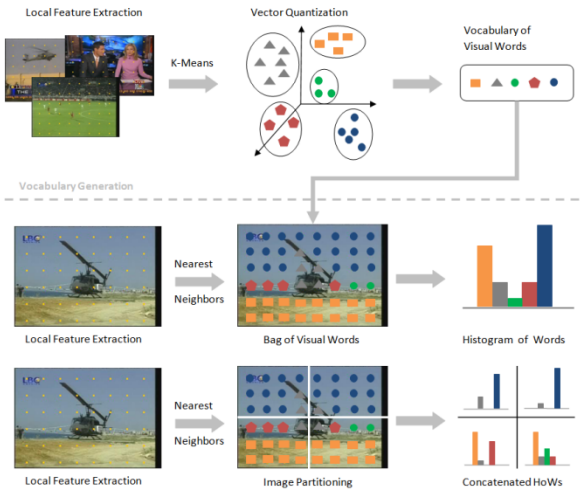

Figure 1: BoW-based image representations.

Since all geometric information gets lost during histogram generation, we additionally used concatenated local histograms to preserve global spatial arrangements (see Figure 1). We applied an spatial image partitioning of 2x2 regions resulting in a 4000-dimensional feature vector.

2.1.1 Soft-Weighting

To consider the similarity of keypoints to the vocabulary entries, the soft-weighting scheme of Jiang et al. [6] was applied during histogram generation. Instead of mapping a keypoint only to its nearest neighbor, the top K nearest visual words were selected. Using a visual vocabulary of N visual words, the importance of a visual word t in the image is represented by the weights of the resulting histogram bins $w = [w_1, \ldots, w_t, \ldots, w_N]$ with

$$w_t = \sum_{i=1}^{K} \sum_{j=1}^{M_i} \frac{1}{2^{i-1}} sim(j,t) \qquad (1)$$

where M_i is the number of keypoints whose i-th nearest neighbor is the visual word t. The Euclidean distance was employed for the comparison of keypoint descriptors and the distance values were transformed to similarities using the following similarity function:

$$sim(x,y) = e^{-\frac{2}{\gamma} d(x,y)} \qquad (2)$$

where d is the Euclidean distance and γ is the maximum distance between two codebook entries. In a postprocessing step, each histogram was normalized by its l_1-norm.

2.1.2 Color Information

Color information was integrated using RGB-SIFT. The SIFT descriptors were computed independently for the three channels of the RGB color model. The final keypoint descriptor is the concatenation of the individual descriptors, resulting in a 3x128-dimensional feature vector. Due to the normalizations during the SIFT feature extraction, the RGB-SIFT descriptor is equal to the transformed color SIFT descriptor, and is therefore invariant against light intensity and color changes or shifts, respectively [16].

2.2 Object-based Features

State-of-the-art object detection approaches [3][18] are utilized to find object appearances for the following 21 object classes:

- "aeroplane"
- "bicycle"
- "bird"
- "boat"
- "bottle"
- "bus"
- "car"
- "cat"
- "chair"
- "cow"
- "dining table"
- "dog"
- "horse"
- "motorbike"
- "person"
- "potted plant"
- "sheep"
- "sofa"
- "train"
- "tv-monitor"
- "face"

Due to the large amount of video data (263569 shots), we abstained from building object sequences and concentrated on the keyframes. We used the Viola-Jones detector [18] for faces and an approach based on deformable part models [3] for the remaining object classes. Using these object detectors trained on separate public data sets, shot-based confidence scores as well as further derived features were computed.

2.1.1 Deformable Part Models

Compared to the last year's system, we used an enhanced version of the object detection approach provided by Felzenswalb et al. [3] using cascades [4], which is more than one order of magnitude faster. The object models [5] were released in conjunction with the PASCAL Visual Object Classes (VOC) Challenge 2010 [2]. The approach uses discriminatively trained mixtures of deformable part models, which consist of a global template that covers the whole object, several smaller part templates, and a model describing the spatial arrangement of the smaller parts. The templates are based on histograms of gradient features. Each object detector delivers a number of bounding boxes and associated confidence scores per shot. The detection threshold was set very low to obtain more bounding boxes per shot. Based on these object detection results, shot-based average and maximum confidence scores were calculated for each object class. In case of no detection result per shot for a specific object class, the average as well as the maximum value was set to the detection threshold.

2.1.2 Viola-Jones Face Detector

In addition to the previously described approach, frontal faces were detected using the face detector provided by the OpenCV library [11]. The face detection approach is an implementation of the approach suggested by Viola and Jones [18] with Lienhart's extensions [9]. The Adaboost-based approach of Viola and Jones was chosen since it is a very fast approach that nearly operates in real-time on today's computers. Since this approach usually reports many detections for a face of slightly different sizes and positions, an average rectangle was computed based on the reported detections, and the number of detections was used as a confidence score.
For each shot, we used the number of faces, the average confidence score, the maximum confidence score, the average size and the maximum size of the detected bounding boxes as features.

2.3 Global Features

Furthermore, we extracted global color and texture features. Therefore, we used the Gnu Image Finding Tool (GIFT) [www.gnu.org/software/gift] to build color and Gabor histograms. The colors were described in the HSV color space. For the color histogram, 18 bins were chosen to represent the hue component, three for the saturation and three for the brightness component. Four additional grey values were used for a better adaption to the human color perception. In total, each color histogram results in a 166-dimensional feature vector.
Global texture characteristics were described using Gabor wavelets. The functions to compute the wavelet coefficients can be expressed as follows [8]:

$$
\begin{aligned}
g_{\theta,\lambda,\varphi,\sigma,\gamma}(x,y) &= e^{-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}} \cos(2\pi\frac{x'}{\lambda}+\varphi) \\
x' &= x\cos\theta + y\cos\theta \\
y' &= -x\sin\theta + y\cos\theta
\end{aligned}
\qquad (3)
$$

A Gabor wavelet is controlled by five parameters: orientation θ, wave length λ, phase φ, radius σ of the Gaussian function, and the aspect ratio γ. The radius of the Gaussian function is chosen proportionally to the wave length, and the aspect ratio is fixed to 1. Gabor energies of a pixel for the different orientation and spatial-frequency combinations were obtained by a superposition of the phases 0 and π/2. We extracted Gabor wavelet features for four orientations and three frequencies. The resulting 12 Gabor energies per pixel were summarized in a Gabor histogram describing the whole image. By distinguishing ten energy classes, we obtained a 120-dimensional Gabor histogram.

3. Multiple Kernel Learning

Kernel-based soft maximum-margin classifiers also called support vector machines (SVM) have proven to be powerful for classifying semantic concepts. In a one-vs.-rest setting we built a support vector machine for each semantic concept. The kernel function of a SVM intuitively measures the similarity between two data instances. We applied the radial basis function (rbf) kernel

$$k_{rbf}(x,y) = e^{-\gamma\|x-y\|^2} \quad (4)$$

and the chi2 kernel

$$k_{\chi^2}(x,y) = e^{-\gamma\chi^2(x,y)} \quad (5)$$

which is based on the corresponding histogram distance

$$\chi^2(x,y) = \sum_i \frac{(x_i-y_i)^2}{x_i+y_i}. \quad (6)$$

While the rbf kernel is used for object-based features, we used the χ^2 kernel for histogram representations.
Different feature representations usually capture only one aspect of the data. Depending on the semantic concept we want to capture, different aspects that are more or less important were considered. Instead of using cross-validation to choose the best performing feature combination, MKL is applied to find an optimal convex kernel combination

$$k = \sum_{i=1}^{n} \beta_i k_i \quad with\ \beta_i \geq 0, \sum_{i=1}^{n} \beta_i = 1 \quad (7)$$

where each kernel k_i takes a different feature representation into account. The optimized kernel weights provide useful information about the relevance of features for the discrimination of semantic concept classes. Besides the l_1-norm MKL, we also investigated non-sparse MKL using the l_2-norm, which leads to a more uniform distribution of kernel weights.
Throughout our experiments, we used the Multiple Kernel Learning framework provided by the Shogun library [15] in combination with the support vector machine implementation of Joachims [7], called SVM light.

4. Experimental Results

In this section, we present our results for the semantic indexing task.
We submitted four full submission runs of category "A". The experiments were evaluated by the TRECVID team [14] based on the inferred average precision measure suggested by Aslam et al. [1]. Figure 2 shows the results of all our submitted runs in terms of mean inferred average precision.
This year, the BoW approach served as a basis for our experiments (Marburg1). We combined histograms-of-visual-words for the whole image with concatenated histograms for an 2x2 image partitioning. The kernel weights of both representations were learned using l_1-norm MKL.
In a first experiment (Marburg2), we added three object-based feature representations to the MKL framework of our baseline system. These object-based feature representations include the face related features, and average and maximum confidence scores, for the remaining 20 object classes, taken into account by using RBF kernels. This approach considering additional object-based features significantly improved our baseline system from 5.58% to 6.29% mean inferred average precision.
In a second experiment, we investigated the impact of the non-sparse l_2-norm MKL, which results in a more uniform distribution of kernel weights (see Figure 3). This run further improved the performance and achieved 6.96% mean average precision, which is our best overall result for the semantic indexing task. In particular, the concepts "animal", "bicycling", "bus", "vehicle" and "ground_vehicle" profited from the additional object-based features and were partly increased by more than 100% (see Figure 4).
In comparison to other teams we achieved the best result for the concepts "vehicle" with 20.1% inferred average precision and "ground_vehicle" with 20.2%. Only one team submitted better results for the concept "cheering" and only two teams for the concepts "bicycling" and "animal".
In the last experiment, we supplemented our feature set with additional global features. A color as well as a

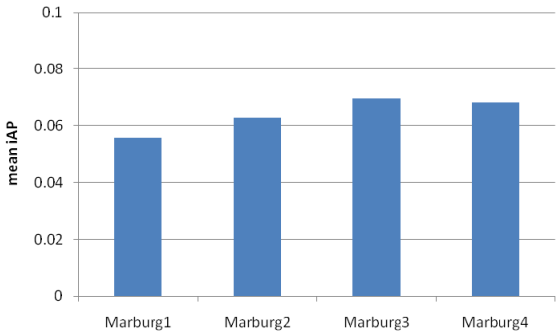

Figure 2: Overview of the results of our four runs in terms of mean inferred average precision.

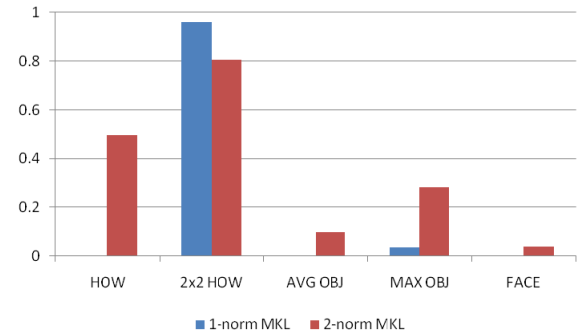

Figure 3: Kernel weights for l_1-norm respectively l_2-norm MKL.

Gabor histogram representation was taken into account by χ^2 kernels. The kernel weights were again learned using non-sparse MKL. This combination of local, global and object-based features achieved no performance gain compared to the previous system. While the concepts "flowers", "cheering", "nighttime", "demonstration_or_protest" and "doorway" were improved by the additional global features, several other concepts like "animal" or "bicycling" dropped. It seems that especially concepts describing scenes profited from global color and texture information.

5. Conclusions

In this paper, we presented our experiments for the semantic indexing task. Based on the success of object-based features in our last year's system, we incorporated further specialized object detectors trained on separate public data sets. A state-of-the-art BoW approach serving as baseline system was supplemented by the resulting object-based features. Instead of just concatenating the different feature representations in an early fusion scheme, we used MKL to find the best feature weighting.

The experiments revealed that the approaches employing additional object-based features significantly improved the overall performance. Some concepts like "animal", "bicycling" or "vehicle" were improved by more than 100%. For "vehicle" and "ground_vehicle", we obtained the best results with 20.1% and 20.2%, respectively, in terms of inferred average precision among all submitted runs. Furthermore, we showed that a more uniform distribution of kernel weights achieved better results than using sparse l_1-norm MKL.

Finally, our best run combining BoW- and object-based features using the l_2-norm MKL obtained a mean inferred average precision of 6.96%.

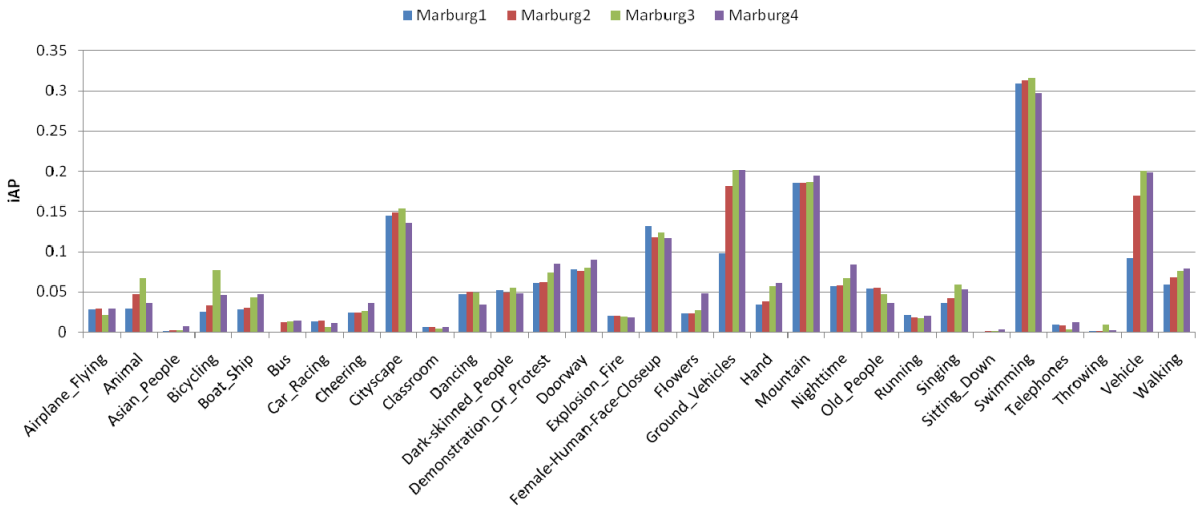

Figure 4: Comparison of our four runs on the plain concept set in terms of inferred average precision.

6. Acknowledgements

This work is financially supported by the German Research Foundation (DFG, PAK 509, Project MT) and the German Ministry of Education and Research (BMBF, D-Grid Initiative, Project MediaGrid).

7. References

1. Aslam, J. A., Pavlu, V., and Yilmaz, E. Statistical Method for System Evaluation Using Incomplete Judgments. In *Proceedings of the 29th ACM SIGIR Conference*, Seattle, 2006, pp. 541-548.
2. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., and Zisserman, A. The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results. www.pascal-network.org/challenges/VOC/voc2008/.
3. Felzenszwalb, P., Girshick, R., McAllester, D., and Ramanan, D. Object Detection with Discriminatively Trained Part-Based Models, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010, pp. 1627-1645.
4. Felzenszwalb, P., Girshick, R., McAllester, D. Cascade Object Detection with Deformable Part Models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2241-2248.
5. Felzenszwalb, P., Girshick, R., and McAllester, D. Discriminatively Trained Deformable Part Models, Release 4, http://people.cs.uchicago.edu/~pff/latent-release4.
6. Jiang, Y., Ngo, C., and Yang, J. Towards Optimal Bag-of-Features for Object Categorization and Semantic Video Retrieval. In *Proceedings of the 6th ACM Int'l Conference on Image and Video Retrieval*, Amsterdam, The Netherlands, 2007, pp. 494-501.
7. Joachims, T. Text Categorization With Support Vector Machines: Learning With Many Relevant Features, In *Proceedings of the 10th European Conference on Machine Learning*, Springer, 1998, pp. 137-142.
8. Kruizinga, P. and Petkov, N. Non-linear operator for oriented texture, *IEEE Transactions on Image Processing*, 8 (10), 1999, pp. 1395-1407.
9. Lienhart, R., Liang, L., and Kuranov, A. A Detector Tree of Boosted Classifiers for Real-time Object Detection and Tracking. In *Proceedings of IEEE Int'l Conference on Multimedia & Expo,* , Vol. 2, Baltimore, Maryland, USA, 2003, pp. 277-280.
10. Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60, 2, 2004, pp. 91-110.
11. OpenCV, Open Computer Vision library, http://sourceforge.net/projects/opencvlibrary.
12. Mühling, M., Ewerth, R., and Freisleben, B. Improving Semantic Video Retrieval via Object-Based Features, In *Proceedings of the 3rd IEEE Int'l Conference on Semantic Computing*, Berkeley, USA, 2009, pp. 109-115.
13. Mühling, M., Ewerth, R., Stadelmann, T., Shi, B., and Freisleben, B. University of Marburg at TRECVID 2009: High Level Feature Extraction. In *Online Proceedings of TRECVID Conference Series 2009*: http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs. org.html.
14. Smeaton, A. F., Over, P., and Kraaij, W. Evaluation campaigns and TRECVid. In *Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, Santa Barbara, California, USA, 2006, pp. 321-330.
15. Sonnenburg, S., Rätsch, G., Schäfer, C., and Schölkopf, B. Large Scale Multiple Kernel Learning, *Journal of Machine Learning Research*, 2006, pp. 1531-1565.
16. Van de Sande, K., Gevers, T., and Snoek, C. A Comparison of Color Features for Visual Concept Classification, In *Proceedings of the ACM 2008 International Conference on Content-Based Image and Video Retrieval*, 2008, pp. 141-150.
17. Vedaldi, A. and Fulkerson, B. VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org, 2008.
18. Viola, P. and Jones, M. J. Robust Real-Time Face Detection. In *International Journal of Computer Vision*, 57(2), Kluwer Academic Publishers, Netherlands, 2004, pp. 137-154.