Weighting Informativeness of Bag-of-Visual-Words by Kernel Optimization for Video Concept Detection

Feng Wang
East China Normal University
Shanghai 200241, P. R. China
fwang@cs.ecnu.edu.cn

Bernard Merialdo
Institute Eurecom
Sophia-Antipolis, France
Bernard.Merialdo@eurecom.fr

ABSTRACT

Bag-of-Visual-Words (BoW) feature has been demonstrated effective and widely used in video concept detection due to its discriminative ability by capturing the local information in images. In the current approaches, all the words in the visual vocabulary are treated equally for the detection of different concepts. This cannot highlight the concept-specific visual information, and thus limits the discriminative ability of BoW feature. In this paper, we propose an approach to boost the performance of video concept detection based on BoW. This is achieved by assigning different weights to the visual words according to their informativeness for the detection of different concepts. Kernel alignment score (KAS) is used to measure the discriminative ability of SVM kernels, and the visual words are weighted as a kernel optimization problem. We show that the SVMs based on weighted visual words with our approach outperform the uniformly weighting and TF-IDF weighting schemes, and the MAP for the 20 concepts from TRECVID 2009 high-level feature extraction is significantly improved.

Categories and Subject Descriptors

I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding—*Video analysis*; I.5.4 [**Pattern Recognition**]: Applications—*Computer vision*.

General Terms

Algorithms, Experimentation, Performance

Keywords

Bag-of-Visual-Words, Kernel Optimization, Concept Detection

1. INTRODUCTION

As a basic step for content-based image and video retrieval, image/video concept detection has been intensively studied in the past few years especially due to the great efforts of TRECVID Workshop [8]. Different approaches and features have been proposed. For classifiers, SVM (Support Vector Machine) has been widely used. Features include global feature, local feature and audio feature. Among these features, Bag-of-Visual-Words (BoW) [1, 4] has achieved great success due to its efficiency and effectiveness by capturing discriminative local image information.

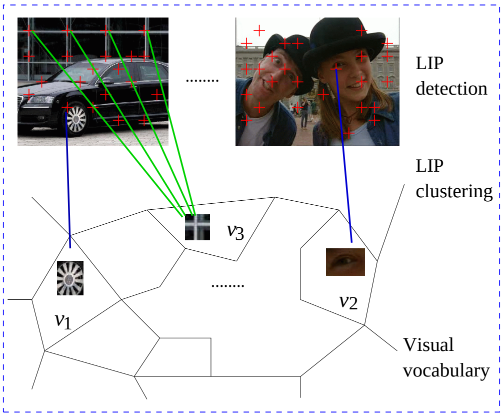

Figure 1: Construction of visual vocabulary.

For the generation of BoW feature, a visual vocabulary is first constructed on a set of training images as illustrated in Figure 1. In each image, the local interest points (LIPs) are detected [7], and described with SIFT (Scale Invariant Feature Transformation) [6]. The LIPs are then clustered into different groups to form a visual vocabulary. This process actually segments the SIFT descriptor space into different Voronoi cells, each corresponding to a visual word (see Figure 1). To compute the BoW feature vector of a given image, each detected LIP is assigned to one or few nearest visual word(s) [4, 5]. This results in a histogram on the visual vocabulary, which can be used as the input for classifier training and testing.

The typical size of the visual vocabulary is 200 to 5000, and BoW feature captures the visual distribution of an image on the whole SIFT descriptor space. On the other hand, the number of keypoints in each image ranges from tens to thousands. The visual information of a concept actually cannot be evenly distributed over the whole vocabulary. Given a concept, some visual words are more informative or important for the detection of a specific concept, while the others may be noisy. For instance, in Figure 1, the visual word v_1 containing a wheel-like image patch is quite important for detecting the concept *Car* or *Vehicle*, but may not contribute to the detection of concept *Person*. Similarly, the presence of v_2 implies there is a *Person* or *Human_face* with high probability. However, the relation between the concept *Car* and visual word v_2 could be weak.

VLS-MCMR'10, October 29, 2010, Firenze, Italy.
Copyright 2010 ACM 978-1-4503-0166-4/10/10 ...$10.00.

In the current approaches, for different concepts, each visual word is treated equally. The discriminative ability of those informative visual words could be seriously reduced considering two factors: i) The background of the images from general videos is extremely complex, which contains a lot of visual information besides the concept-related objects and scenes. Thus, the informative visual words can be easily noised. ii) In most current datasets, the number of positive examples are limited for many concepts. For instance, in Sound & Vision data used for TRECVID evaluation, there are only tens of positive examples for some concepts. Therefore, denoising is hard without enough data available. This inspires us to select the visual words which are more informative to boost the performance of concept detection.

To weight the importance of visual words to a specific concept, the classical TF-IDF (Term Frequency - Inverse Document Frequency) approach is the first candidate. Although TF-IDF has proved useful in information retrieval, it is not appropriate for weighting the visual words in concept detection. First, DFs of visual words do not convey much useful information. This is because each image contains a lot of noisy information in the background, and each visual word can be found in the positive samples of almost every concept, meaning that DFs for different words are almost the same. Second, TF is also not a good hint for the importance of the visual word. The importance of a visual word for the detection of a concept is dependent on its informativeness instead of its frequency. For instance, in Figure 1, the presence of word v_1 is very important for detecting the car. Although there is only one keypoint assigned to v_1, it is more important compared with those keypoints in the background assigned to v_3. Furthermore, the statistic information computed in TF-IDF approach is usually not reliable due to the lack of positive examples as mentioned above.

In this paper, we propose an approach to measure the importance of each visual word for given concepts. This is achieved by iteratively updating the weights to push the SVM kernel towards the optimal one. For this purpose, kernel alignment score (KAS) is used to measure the discriminative ability of SVM kernel (Section 2). The problem is then to maximize the KAS score by optimizing the weights of different visual words (Section 3). Finally, the resulting weights are used in a modified kernel for concept detection.

2. EVALUATING SVM KERNELS

In current approaches for concept detection, SVM (Support Vector Machine) has been the mostly frequently used. Typically concept detection is treated as a *one* vs. *all* binary classification problem. At the training stage, two classes are manually labelled: positive examples which contain the given concept and negative ones in which the concept is not present. To measure the fitness of the weights of visual words, the performance of SVM classifier is the best hint. However, it is not applicable to evaluate the performance by training, cross-validation and testing from time to time during the weighting procedure.

The performance of SVM is mainly dependent on the ability of kernel matrix to discriminate between positive and negative samples. Different factors can affect kernel matrix such as kernel function format, features, and parameter settings. Kernel optimization is to find a better kernel by optimizing these factors. In this paper, we weight the visual words in the framework of kernel optimization, i.e. we attempt to find the optimal weights that can produce the best kernels.

For SVM, an optimal kernel K^{opt} [2] should satisfy

$$K_{ij}^{opt} = \begin{cases} +1 & \text{if } l_i = l_j \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where i, j are two examples and $l_i = +1$ (or -1) if i is a positive (or negative) example. In K^{opt}, the kernel values between samples with the same labels are maximized, while the values between samples with different labels are minimized. Thus, this optimal kernel can perfectly discriminate between different classes.

However, the actual kernels used in practice are usually not optimal due to the imperfect features and kernel functions. To measure how well a given kernel K is aligned with optimal kernel, the kernel alignment score (KAS) [2] is used

$$\hat{S} = \frac{\sum_{i,j} K_{ij} \cdot l_i \cdot l_j}{N \cdot \sqrt{\sum_{i,j} K_{ij}^2}} \tag{2}$$

where N is the total number of examples. Generally, a kernel with higher KAS score is better at discriminating examples of different classes, and can potentially achieve better performance for classification. In our approach, we employ KAS to measure the fitness of SVM kernel and weight the visual words by maximizing KAS scores.

3. WEIGHTING VISUAL WORDS BY KERNEL OPTIMIZATION

3.1 Problem Formulation

For visual vocabulary construction, we first detect the local interest points (LIPs) with Hessian Laplacian detector [7] on a set of training images. The LIPs are described with SIFT [6] and then clustered into c($c = 500$ in our experiments) visual words by employing k-means algorithm to form a visual vocabulary. To generate BoW feature of a given image i, each detected LIP is assigned to the nearest cluster (or visual word), and the image is represented by a histogram on the vocabulary as $X_i = [x_{i1}, x_{i2}, \cdots, x_{ic}]$.

The resulting BoW features are then used for SVM training and classification. In our approach, we adopt RBF (Radial Bias Function) kernel which is the most frequently used for SVM classification. Given two images i, j and their BoW representation X_i, X_j, the RBF kernel is defined as

$$\hat{K}_{ij} = \exp(-\sigma \cdot \sum_{m=1}^{c} (x_{im} - x_{jm})^2) \tag{3}$$

As discussed in Section 1, the existing approaches assign the same weights to all visual words. In this paper, we attempt to measure the importance of visual words for each concept and assign different weights to them accordingly. Here we just consider the detection of one concept and denote the weight vector of visual words as $w = [w_1, w_2, \cdots, w_c]$ where $\sum_{m=1}^{c} w_i = 1$. By considering the weights, Equation 3 can be rewritten as

$$K_{ij} = \exp(-\sigma \cdot \sum_{m=1}^{c} w_m \cdot (x_{im} - x_{jm})^2) \tag{4}$$

where the more important visual words contribute more to the distance measure between examples.

To select a weight vector which can result in better kernels, we employ the kernel alignment score in Equation 2 to measure the discriminative ability of kernels. Equation 2 assumes the two classes are balanced. However, this is not the case in current datasets, where there are usually many more negative examples than positive ones. This may bias the resulting KAS towards the negative class. To deal with this imbalance problem of the datasets, we modify Equation 2 by assigning different weights to positive and negative examples as follows

$$\alpha_i = \begin{cases} 1 & \text{if } l_i = -1 \\ \frac{N^-}{N^+} & \text{otherwise} \end{cases} \tag{5}$$

where N^- and N^+ are the numbers of negative and positive examples in training data respectively. Equation 2 is then modified as

$$S = \frac{\sum_{i<j} K_{ij} \cdot l_i \cdot l_j \cdot \alpha_i \cdot \alpha_j}{N' \cdot \sqrt{\sum_{i<j} \alpha_i \cdot \alpha_j \cdot K_{ij}^2}} \tag{6}$$

where $N' = \sum_{i<j} \alpha_i \alpha_j$. Eventually the problem is formulated as searching for an optimal weight vector w^{opt} such that the KAS score defined by Equations 6 is maximized.

3.2 Gradient-based Weights Optimization

Gradient-descent approach is widely used for optimization. In [3], gradient-based algorithm is employed to select SVM parameters for bacterial gene start detection in biometrics. In our approach, we weight the visual words by adopting a similar gradient-descent algorithm to optimize the SVM kernels by maximizing the KAS in Equation 6. Based on Equation 6, we calculate the partial derivative of S to the weight w_m as

$$\frac{\partial S}{\partial w_m} = \sum_{i<j} \frac{\partial S}{\partial K_{ij}} \cdot \frac{\partial K_{ij}}{\partial w_m} \tag{7}$$

$$\frac{\partial K_{ij}}{\partial w_m} = K_{ij} \cdot (-\sigma \cdot (x_{im} - x_{jm})^2) \tag{8}$$

Besides the weights of visual words, we also optimize σ in Equation 4 which is an important parameter for SVM kernels

$$\frac{\partial S}{\partial \sigma} = \sum_{i<j} \frac{\partial S}{\partial K_{ij}} \cdot \frac{\partial K_{ij}}{\partial \sigma} \tag{9}$$

$$\frac{\partial K_{ij}}{\partial \sigma} = K_{ij} \cdot (-\sum_{m=1}^{c} w_m \cdot (x_{im} - x_{jm})^2) \tag{10}$$

In Equation 8, the weights of different visual words are supposed to be independent on each other. According to our experiment, this assumption is reasonable. Strictly speaking, there might be some weak correlations between the importance of different visual words. For instance, two visually similar words may have the similar weights.

Based on Equations $7 - 10$, , we iteratively update the weight vector w of visual words so as to maximize the kernel alignment score in Equation 6. Below is the algorithm for optimization:

1. *Initialize* $w_m = 1/c$ *for* $m = 1, 2, \cdots, c$ *and* $\sigma = \sigma_0$. *Calculate the initial KAS score* S *by Equation 6.*
2. *For each weight* w_m *and* σ, *calculate the partial derivative* $\frac{\partial S}{\partial w_m}$ *and* $\frac{\partial S}{\partial \sigma}$ *by Equations 7-10.*
3. *Update weights* $w'_m = w_m \cdot (1 + sign(\frac{\partial S}{\partial w_m}) \cdot \delta_w)$ *and* $\sigma' = \sigma \cdot (1 + sign(\frac{\partial S}{\partial \sigma}) \cdot \delta_\sigma)$, *where* $sign(t) = \begin{cases} +1 & if \;\; t > 0 \\ 0 & if \;\; t = 0 \\ -1 & if \;\; t < 0 \end{cases}$. δ_w *and* δ_σ *are two constants to be determined. Get the new weights by normalizing* $w_m = \frac{w'_m}{\sum_{k=1}^{c} w'_k}$.
4. *Calculate the new kernel alignment score* S' *using the updated weights and* σ. *If* $\frac{S'-S}{S} < thres$, *stop; otherwise,* $S = S'$ *and go to step 2.*

In step 1, $\sigma_0 = N' / \sum_{i<j} \sum_{m=1}^{c} w_m \cdot (x_{im} - x_{jm})^2$ which is the inverse of the average distance between examples, and a good empirical choice of σ for SVM paramter selection. In step 3, the weight vector (and σ) is updated by a small value δ_w (and δ_σ). For the determination of δ_w (and δ_σ), a larger value can push the weights (and σ) to the optimal one at higher speed at the beginning. However, this risks missing the local minimal point by skipping a large distance in each step and cannot find the optimal weights. A smaller δ_w (and δ_σ) can avoid this problem, but it takes more iterations to converge. In our experiments, we empirically set $\delta_w = 0.02$ and $\delta_\sigma = 0.1$. In step 4, *thres* is set to be 0.5% so as to stop the optimization when the improvement on KAS becomes minor. After the optimization process, the resulting weight vector is used to train SVMs with the kernel defined in Equation 4 for concept detection.

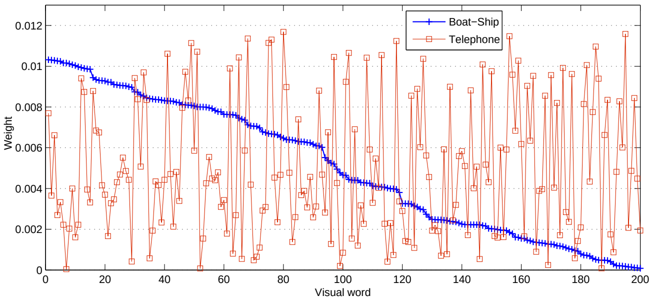

Figure 2: Weights of visual words for two concepts.

4. EXPERIMENTS

The experiments are carried out on the development and test sets of TRECVID 2007 Sound & Vision data. There are totally 21532 and 22084 keyframes for development and testing respectively, which are extracted from about 100-hour videos. The twenty concepts (see Table 1) from TRECVID 2009 high-level feature extraction task are detected. The definitions of these concepts can be found at [8].

4.1 Concept-Specific Weights of Visual Words

Figure 2 plots the computed weights of visual words for two concepts: *Boat_Ship* and *Telephone* by our approach described in Section 3 (for the convenience of illustration, we use a small vocabulary with 200 visual words). In Figure 2, the visual words are sorted in descending order of their weights for concept *Boat_Ship*, and the weights for *Telephone* (red marks) are then plotted accordingly for comparison.

From Figure 2, we can see: i) For one given concept (*Boat_Ship*), some visual words are assigned with much larger weights than others. This demonstrates the variations of different visual words' informativeness for detecting the concept. Therefore, it is important to select the most informative visual words in order to improve the discriminative ability of the classifiers. ii) By comparing different concepts, the weights for the same visual word are also quite different. One visual word which is important for a given concept may not be important for another one. Thus, it is necessary to weight the visual words for different concepts instead of assigning the same weights to all concepts.

4.2 Comparison with TF-IDF Weighting and Uniformly Weighting Approaches

We compare our approach with uniformly weighting and TF-IDF weighting approaches. For uniformly-weighting scheme, all visual words are assigned with the same weights as used in existing approaches.

TF-IDF (Term Frequency-Inverse Document Frequency) is a classical weighting algorithm which is widely used in information retrieval. Given a concept t, all the samples containing t compose a document $D_t = \{X_i | l_i = +1 \ for \ concept \ t\}$. For each visual word v, the Term Frequency is calculated as

$$tf_{vt} = \frac{F_{vt}}{\sum_{v'} F_{v't}} \qquad (11)$$

where F_{vt} is the occurrence frequency of word v in document D_t, and the normalization factor $\sum_{v'} F_{v't}$ is the total occurrence frequency of all visual words in D_t.

Inverse Document Frequency is calculated as

$$idf_v = \log(\frac{M}{|\{D_{t'} | F_{vt'} \neq 0\}|} + 1) \qquad (12)$$

where M is the total number of documents (concepts), and $|\{D_{t'} | F_{vt'} \neq 0\}|$ is the number of documents that contain the word v. Thus, the importance of v for concept t can be weighted by

$$\hat{w}_{vt} = tf_{vt} \cdot idf_v \qquad (13)$$

The resulting weights are then used to compute the kernel in Equation 4 for SVM training and classification.

In our experiments, we adopt Average Precision (AP) to evaluate the performance of concept detection. Two-fold cross-validation is carried out on development and test set. Table 1 shows the performances of concept detection by three weighting schemes. For TF-IDF approach, minor improvement (0.58%) on the MAP (Mean Average Precision) for 20 concepts is observed compared with uniformly-weighting approach. For different concepts, the performance improvement by TF-IDF is inconsistent. As discussed in Section 1, TF-IDF is not a good hint for the informativeness of visual words in concept detection. The limited number of positive examples and the noises in images reduce the reliability of the statistics information used by TF-IDF.

Table 1: Comparison between different weighting schemes.

Concept	Uniform	TF-IDF		Our approach	
	AP	AP	Improve	AP	Improve
Airplane_flying	0.1177	0.1145	-2.72%	0.1315	**11.72%**
Boat_Ship	0.1698	0.1645	-3.08%	0.1925	**13.37%**
Bus	0.0086	0.0092	**6.98%**	0.0117	**36.05%**
Chair	0.0403	0.0420	4.22%	0.0440	9.18%
Cityscape	0.0845	0.0843	-0.24%	0.0920	8.88%
Classroom	0.0100	0.0096	-4.00%	0.0122	**22.00%**
Demonstration	0.0256	0.0277	**8.20%**	0.0280	9.38%
Doorway	0.0836	0.0864	3.35%	0.0907	8.49%
Female_Face_Closeup	0.0762	0.0752	-1.31%	0.0793	4.07%
Hand	0.0932	0.0920	-1.29%	0.0995	6.76%
Infant	0.0112	0.0120	**7.14%**	0.0117	4.46%
Nighttime	0.1326	0.1315	-0.83%	0.1354	2.11%
People-dancing	0.0248	0.0258	4.03%	0.0263	6.05%
Person-eating	0.2673	0.2662	-0.41%	0.2687	0.52%
Playing-music	0.0899	0.0987	**9.79%**	0.0976	8.57%
Person-playing-soccer	0.0740	0.0721	-2.57%	0.0825	**11.49%**
Person-riding-bicycle	0.2799	0.2841	1.50%	0.2894	3.39%
Singing	0.0284	0.0278	-2.11%	0.0310	9.15%
Telephone	0.0213	0.0232	**8.92%**	0.0268	**25.82%**
Traffic-intersection	0.3158	0.3197	1.23%	0.3214	1.77%
MAP	0.0977	0.0983	**0.58%**	**0.1036**	**6.01%**

Table 1 also compares our approach with uniformly weighting scheme. Overvall an improvement of 6.01% is achieved on MAP. Large margins on the performance of two approaches can be observed for some concepts including *Bus* (36.05%), *Telephone* (25.82%), and *Classroom* (22%). These concepts are mostly object-related with specific exterior appearance. By assigning larger weights to the visual words describing their appearances, the detection of these concepts can be improved. Furthermore, the improvement is consistent for different concepts. This is because our weighting scheme aims to optimizing the discriminative ability of SVM kernels, and thus improving the performance of concept detection. Note that for concepts *Person-eating*, *Riding-bicycle* and *Traffic-intersection*, the performance of uniformly weighting approach is already quite good and the improvement with our approach is insignificant. This is because there are many duplicate examples in training and testing sets, which almost dominate the APs for these concepts, and to some extent, distort the improvement on MAP.

For speed efficiency, all the partial derivatives in Equations 7-11 can be calculated by scanning each pair of examples for one time. In our experiments, one iteration takes around $10-15$ minutes. The whole optimization process can be finished in around 5 hours for most concepts. The training and classification of SVM with weighted visual words do not take extra time compared with original approaches. To speed up the optimization process, random sampling on the negative examples can be used without much information loss since there are more than enough negative ones.

5. CONCLUSION

In this paper, we have presented our approach for video concept detection with concept-specific weights of Bag-of-Visual-Words. To the best of our knowledge, this is the first algorithm to measure the informativeness of visual words in BoW based concept detection. Given a concept, the weights of different visual words are calculated in the framework of SVM kernel optimization. By assigning different weights to the visual words according to their importance, the discriminative ability of SVM kernel is strengthened and thus the performance of concept detection is improved. In our experiments, SVM classifier with RBF kernel is employed. The proposed approach could be extended to select the optimal weights for other kernels and classifiers.

6. REFERENCES

[1] S. F. Chang and *et. al*, "Columbia University/VIREO-CityU/IRIR TRECVID2008 High-Level Feature Extraction and Interactive Video Search", *NIST TRECVID Workshop*, 2008.

[2] N. Cristianini, J. Kandola, A. Elisseeff, and J. S-Taylor, "On Kernel Target Alignment", *Advances in Neural Information Processing Systems*, vol. 14, pp. 367-373, 2002.

[3] C. Igel, T. Glasmachers, B. mersch, N. Pfeifer, and P. Meinicke, "Gradient-based Optimization of Kernel-Target Alignment for Sequence Kernels Applied to Bacterial Gene Start Detection", *IEEE/ACM Trans. on Computational Biology and Bioinformatics*, vol. 4, no. 2, pp. 216-226, 207.

[4] Y. G. Jiang, C. W. Ngo, and J. Yang, "Towards Optimal Bag-of-Features for Object Categorization and Semantic Video Retrieval", *Conf. on Image and Video Retrieval*, 2007.

[5] Y. G. Jiang and C. W. Ngo, "Bag-of-Visual-Words Expansion Using Visual Relatedness for Video Indexing", *ACM SIGIR*, 2008.

[6] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", *Int. Journal of Computer Vision*, vol. 60, 2004.

[7] K. Mikoljczyk and C. Schmid, "Scale and affine invariant interest point detectors," *Int. Journal of Computer Vision*, vol. 60, pp. 63–86, 2004.

[8] TRECVID workshop.
http://www-nlpir.nist.gov/projects/trecvid/.