論文 / 著書情報
Article / Book Information

Title	MULTIMEDIA EVENT DETECTION USING GMM SUPERVECTORS AND SVMS
Author	Yusuke Kamishima, Nakamasa Inoue, Koichi Shinoda, Shunsuke Sato
Journal/Book name	ICIP 2012, , , pp. 3089-3092
Issue date	2012, 10
DOI	http://dx.doi.org/10.1109/ICIP.2012.6467553
URL	http://www.ieee.org/index.html
Copyright	(c)2012 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other users, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works for resale or redistribution to servers or lists, or reuse of any copyrighted components of this work in other works.
Note	このファイルは著者（最終）版です。 This file is author (final) version.

MULTIMEDIA EVENT DETECTION USING GMM SUPERVECTORS AND SVMS

Yusuke Kamishima[1], *Nakamasa Inoue*[1], *Koichi Shinoda*[1], *Shunsuke Sato*[2]

[1]Department of Computer Science, Tokyo Institute of Technology
[2]Digital Platform Technology Development Headquarters, Canon Inc.

ABSTRACT

In multimedia event detection, complex target events are extracted from a large set of consumer-generated videos taken in unconstrained environments. We devised a multimedia event detection method based on GMM supervectors and support vector machines (SVMs) using multiple features. A GMM supervector consists of the parameters of a Gaussian mixture model (GMM) for the distribution of local features extracted from a video clip. A GMM is regarded as an extension of the Bag-of-Words (BoW) to a probabilistic framework, and thus, it can be expected to be robust against the data insufficiency problem. This method outperformed previous methods including BoW in experiments using the dataset of the multimedia event detection task in TRECVID2010 and 2011.

Index Terms— Multimedia event detection, Feature extraction, GMM-Supervector, Support vector machines

1. INTRODUCTION

The amount of consumer-generated videos we can access over the Internet has been rapidly increasing. Since it has become difficult to manage them manually, there is a need for automatic methods to search them. In particular, detecting *events* depicted in a video enables us to get significant information. Here, events are characterized by human motions and actions that are unusual in daily life. Examples of such unusual events are varied and they include, for example, birthday parties, goals in a soccer game, and people meeting in a place under surveillance.

Most studies have been aimed at identifying events in professionally produced videos such as sports [1] and movies [2], or at surveillance videos taken from fixed camera views [3]. These studies used event-specific methods which rely heavily on the temporal-spatial structures of the events.

On the other hand, consumer videos are often made in unconstrained environments using various recording devices. The images may include unsteady camera motions, and they are often edited haphazardly. Because of these qualities, most of the previous methods cannot be directly applied to them.

Several studies have been done on event detection in consumer videos. For example, Ke et al. [4] used a volumetric feature framework, which converts optical flows into 3D features. Niebles et al. [5] proposed a human action categorization method using spatio-temporal features. However, the targets of these studies are rather simple events such as "walking", "running", or "handwaving".

Our purpose is to detect a complex event occurring at a specific place and time and consisting of a number of human activities from a large amount of consumer-generated videos at the clip level. For instance, the event "birthday party" consists of "person", "cake", "decoration", "singing", or captions including the word "birthday". The problem here is that it is difficult to provide a sufficient amount of training data to learn the features of each event. This *data insufficiency* problem occurs in many pattern recognition applications.

A few Bag-of-Words (BoW) based methods have been proposed [6, 7] for detecting these complex events, and they have proved to be effective. Since these methods use hard clustering, quantization errors degrade detection performance, and they may not deal well with *unseen* features when the amount of training data is small. The semantic event model (SM) [7], which models the relationship of objects and/or activities making up an event, has also been used for this purpose. However, it is difficult to learn semantic relations between objects and/or activities with a small amount of data. We need an event model whose parameters can be robustly estimated when the amount of training data is small.

In this paper, we propose an event detection method based on Gaussian mixture models (GMMs) supervectors and support vector machines (SVMs) using multiple local features. A GMM represents the distribution of local features extracted from a video clip. Our GMM-based method can be regarded as an extension of the BoW methods to a probabilistic framework, and thus, has less quantization errors. It can be expected to model events more precisely with smaller amounts of data in comparison with the BoW-based method. The use of multiple local features, including visual, audio, and temporal features, is expected to enhance detection performance. SVMs discriminate events and non-events precisely even with a small amount of training samples. On the other hand, our method does not explicitly utilize the global temporal-spatial features of each event, which are difficult to model with a small amount of data. We expect that, even without them, the combination of GMMs, multiple features, and SVMs will have high detection performance.

The idea of combining of GMMs and SVMs (GSSVM) was first proposed for speaker verification [8]. It has since been applied to video recognition, in particular, the object

detection [9]. Event detection using GMMs of SIFT features [10] in a single shot was proposed by Zhou et al. [11]. To the best of our knowledge, we are the first to apply this framework to the detection of complex events in video clips consisting of many shots.

This paper is organized as follows. Section 2 explains the local features we used. Section 3 explains the GMM supervector and how we use it. Section 4 describes the experimental results and their analysis. We conclude in Section 5.

2. FEATURE EXTRACTION

Since videos have multi-modality, it is important to use multiple features to build a high-accuracy multimedia event detection system. We used five types of features that complement each other; three types of visual features with sparse sampling or dense sampling, audio features, and spatio-temporal features. We sampled visual features from one video frame every two seconds in order to reduce computational costs. Audio features should be effective for events with specific sounds, such as a parade featuring a marching band or orchestra. Spatio-temporal features represent local spatio-temporal changes, occurring often in events containing rapid movement such as dancing or jumping.

1. SIFT with Harris-Affine region detector (SIFT_har)
SIFT (Scale-Invariant Feature Transform) [10], which is invariant to image scaling and changing illumination, is used in image analysis applications. We extract 128-dimensional SIFT features from the Harris-Affine regions.

2. SIFT with Hessian-Affine region detector (SIFT_hes)
We also use SIFT features extracted from the Hessian-Affine regions. The Hessian-Affine region detector is often used to detect blobs and is known to be complementary to the Harris-Affine region detector. The combination of different detectors can improve a method's robustness to noise.

3. Audio MFCC features
Audio is an important clue when analyzing video content. We use MFCC (Mel Frequency Cepstral Coefficient) features, which are often used in speech recognition. In addition, we use ΔMFCC, ΔΔMFCC, Δpower, and ΔΔpower. The dimension of a feature vector is 38. We compute the MFCC feature over a 24-ms time window with a 12-ms overlap.

4. Spatio-temporal features
Features extracted from STIPs (Space-Time Interest Points) [12] are expected to be effective in video recognition because STIPs are the regions where spatial changes and temporal changes are large. We extract 72-dimensional HOG (Histograms of Oriented Gradient) features and 90-dimensional HOF (Histograms of Optical Flow) features from one STIP and combine these two vectors.

5. HOG features with dense sampling
We also use 34-dimensional HOG [13] features sampled densely from an image. The dense sampling of HOG computationally costs less than SIFT does. A vector consists of 8-bin histograms of gradients extracted from 2×2 blocks and the coordinates of the center of the blocks in image. Different from features based on keypoints such as SIFT or STIP, dense sampling gives us a fixed number of features, although they may include some noise.

3. GMM SUPERVECTOR AND SVM

In event detection, we first make a Gaussian mixture model (GMM) for each of the five feature types. Then, we construct a Gaussian supervector from each GMM by using MAP adaptation and use it as an input for a SVM classifier. Finally, we fuse the outputs of the SVMs for the five feature types and use the result as the detection score. We explain each step below.

3.1. Gaussian Mixture Models

A Gaussian mixture model (GMM), whose probability density function is given by

$$p(x|\theta) = \sum_{k=1}^{K} w_k \mathcal{N}(x|\mu_k, \Sigma_k), \tag{1}$$

is used to model a video clip. Here, x is a feature vector for one of the five feature types, $\theta = \{w_k, \mu_k, \Sigma_k\}_{k=1}^{K}$ is a set of GMM parameters. K is the number of Gaussian mixture components (vocabulary size), w_k is the weight for mixture component k, and $\mathcal{N}(x|\mu_k, \Sigma_k)$ is a Gaussian probability density function with a mean vector μ_k and a covariance matrix Σ_k for mixture component k.

3.2. MAP adaptation

The GMM parameters are estimated for each clip using the maximum a posteriori (MAP) criterion. This process is often called MAP adaptation [14]. In this adaptation, the parameters of a universal background model (UBM), which are estimated from all video clips, are utilized as the prior distribution for Gaussian means. This adaptation is particularly effective when the amount of data available is small. Let $\theta^{(U)} = \{w_k^{(U)}, \mu_k^{(U)}, \Sigma_k^{(U)}\}_{k=1}^{K}$ be the parameter set of UBM U, where $w_k^{(U)}$ is the weight, $\mu_k^{(U)}$ is the mean vector, $\Sigma_k^{(U)}$ is the covariance matrix for the mixture component k of U. Then, the MAP estimate $\hat{\mu}_k$ for the Gaussian mean is

$$\hat{\mu}_k = \frac{\tau \mu_k^{(U)} + \sum_{i=1}^{n} c_{ik} x_i}{\tau + \sum_{i=1}^{n} c_{ik}}, \tag{2}$$

$$c_{ik} = \frac{w_k^{(U)} \mathcal{N}(x_i|\mu_k^{(U)}, \Sigma_k^{(U)})}{\sum_{k=1}^{K} w_k^{(U)} \mathcal{N}(x_i|\mu_k^{(U)}, \Sigma_k^{(U)})}. \tag{3}$$

where $X = \{x_i\}_{i=1}^{n}$ is a feature vector set of one of the five feature types extracted from a video clip, c_{ik} is the contribution rate of x_i for the k-th Gaussian component (the posterior probability of x_i being at the k-th Gaussian component), and τ is a hyper-parameter which controls the weight of the prior against the maximum likelihood estimate.

3.3. GMM supervector

After MAP adaptation, a GMM supervector $\phi(X)$ is constructed for each video clip by concatenating the mean vectors of all the mixture components in the corresponding GMM as:

$$\phi(X) = (\tilde{\mu}_1^{\mathrm{T}} \tilde{\mu}_2^{\mathrm{T}} \dots \tilde{\mu}_K^{\mathrm{T}})^{\mathrm{T}},\ \tilde{\mu}_k = \sqrt{w_k^{(U)}} (\Sigma_k^{(U)})^{-\frac{1}{2}} \hat{\mu}_k. \quad (4)$$

Here, each mean vector is normalized by its related weight and variance. This GMM supervector is then input to the support vector machine.

3.4. Detection by support vector machines

We use a support vector machines (SVMs) with the following RBF-kernel for each of the five feature types to detect each event:

$$k(X_i, X_j) = \exp(-\gamma \|\phi(X_i), \phi(X_j)\|_2^2), \quad (5)$$

where $\|x\|_2^2$ is the squared 2-norm of x, X_i and X_j are sets of feature vectors and γ is an experimentally optimized control parameter. We set γ to the inverse of the average distance between two GMM supervectors.

3.5. Fusion of the features

We train a SVM for each event and each future. The detection score for the event E is given by

$$s_E(X) = \sum_{\mathrm{F}} \alpha_{E,\mathrm{F}} f_{E,\mathrm{F}}(X) \quad (6)$$

where $f_{E,\mathrm{F}}$ is the discriminative function, which is the output of an SVM trained using feature type F $\in$\{SIFT_har, SIFT_hes, MFCC, STIP, HOG\}, and $\alpha_{E,\mathrm{F}}$ is the fusion weight for E and F. Our combination of GMM supervectors and SVMs is a computationally more efficient approximation of Fisher kernels for GMMs.

4. EXPERIMENT

4.1. Experimental condition

We used the video dataset of the Multimedia Event Detection (MED) task in TRECVID2010 and TRECVID2011 [15]. The TRECVID2010 MED dataset has 3,468 videos, of which 1,744 videos are for training and 1,724 are for testing. The target events are manually annotated and consist of "Assembling a shelter", "Batting a run", and "Making a cake". The positive clips of each event amount to about 50 for training and 50 for testing. The TRECVID2011 MED dataset has 44,904 videos, of which 13,083 videos are for training and 31,821 videos are for testing. Ten target events are listed in Table 2, together with the results. Each event has between 80-230 positive clips for training and testing.

We also used the same evaluation criteria as in the TRECVID MED task. These criteria are mainly based on the missed detection rate (P_{MD}), false alarm rate (P_{FA}), and minimum Normalized Detection Cost (MNDC). The Normalized Detection Cost (NDC) is a linear combination of the probabilities of two types of errors; P_{MD} and P_{FA}. P_{MD}, P_{FA}, and NDC are given by

$$\begin{aligned}
\mathrm{NDC} &= \frac{F_{\mathrm{MD}} + F_{\mathrm{FA}}}{\min(C_{\mathrm{MD}} P_{\mathrm{targ}}, C_{\mathrm{FA}}(1 - P_{\mathrm{targ}}))}, & (7)\\
F_{\mathrm{MD}} &= C_{\mathrm{MD}} P_{\mathrm{MD}} P_{\mathrm{targ}}, & (8)\\
F_{\mathrm{FA}} &= C_{\mathrm{FA}} P_{\mathrm{FA}} (1 - P_{\mathrm{targ}}), & (9)\\
P_{\mathrm{MD}} &= N_{\mathrm{MD}} / N_{\mathrm{pos}}, & (10)\\
P_{\mathrm{FA}} &= N_{\mathrm{FA}} / N_{\mathrm{neg}}, & (11)
\end{aligned}$$

where, C_{MD} and C_{FA} are parameters which control the weights of the missed detection rate and false alarm rate, respectively. N_{MD} is the number of positive videos which are not detected, and N_{FA} is the number of negative videos of E which are detected as positive. N_{pos} is the number of the positive videos, and N_{neg} is the number of the negative videos. P_{targ} is the prior probability of a target event occurring. We also used the same predefined parameters of the task: $C_{\mathrm{MD}} = 80$, $C_{\mathrm{FA}} = 1$, and $P_{\mathrm{targ}} = 0.001$. MNDC is the minimum of NDC over the detection threshold T, which is the value when the detection threshold T is optimized posteriorly.

We set the number of Gaussian components of the GMM to 512 and τ to 20.0 for all the GMMs. These parameters were determined by our preliminary experiments. The fusion weight of each feature, $\alpha_{E,\mathrm{F}}$, was determined by 2-fold cross validation.

4.2. Result and analysis

We compared our method with the previous method proposed by Jiang et al. [7] which combined BoW and semantic event models (SM). It should be noted that it had the best performance in the original TRECVID2010 MED competition. We show the result in Table 1. Since we used different features from theirs, it is difficult to directly compare the performance. Our method achieved mean MNDC 0.558 when we used three features: SIFT_hes, MFCC, and STIP. Their method had mean MNDC 0.579 when they used not only the same three features but also another feature, Difference of Gaussian (DOG) (SIFT_dog in Table 1), which is more likely to detect edges than the Harris-Affine detector and the Hessian-Affine detector. The performance of their method further improved to mean MNDC 0.565 when they additionally used Earth Mover's Distance (EMD) as a metric between two BoW histograms. The performance of our method was significantly better these two results, and was further improved to mean MNDC 0.534 when we added SIFT_har and HOG features. The effectiveness of our method was thus confirmed.

Table 2 lists the results for 10 events in the TRECVID2011 MED task. We can see that P_{MD} is always higher than P_{FA}. This is because the detrimental cost C_{FA} is much larger than

Table 1. Comparison of the proposed methods with the previous method proposed by Jiang et al. [7] on the TRECVID2010 MED dataset. Mean MNDC is the mean of MNDCs taken over the three target events.

Features-Methods	Mean MNDC
SIFT_dog+SIFT_hes+MFCC+STIP-**BoW** [7]	0.586
SIFT_dog+SIFT_hes+MFCC+STIP-**BoW+SM** [7]	0.579
SIFT_dog+SIFT_hes+MFCC+STIP-**BoW+SM+EMD** [7]	0.565
SIFT_hes+MFCC+STIP-**GSSVM**	0.558
SIFT_hes+MFCC+STIP+SIFT_har-**GSSVM**	0.552
SIFT_hes+MFCC+STIP+SIFT_har+HOG-**GSSVM**	0.534

Table 2. Our results in the TRECVID2011 MED task. P_{MD} and P_{FA} are given with the threshold T which gives MNDC.

Event	P_{MD}	P_{FA}	MNDC
Birthday party	0.526	0.009	0.636
Changing a vehicle tire	0.460	0.008	0.556
Flash mob gathering	0.220	0.008	0.324
Getting a vehicle unstuck	0.253	0.014	0.421
Grooming an animal	0.471	0.013	0.639
Making a sandwich	0.514	0.014	0.687
Parade	0.372	0.010	0.500
Parkour	0.289	0.009	0.400
Repairing an appliance	0.308	0.009	0.420
Working on a sewing project	0.580	0.007	0.664

C_{MD} and N_{neg} is much larger than N_{pos}. Accordingly, we have to decrease P_{MD} to get a much lower MNDC. We analyzed the missed positive clips. We found most errors were related the video editing process. Editing means changes of shots, captions, subtitles, or other effects. For example, 47% of the missed positive clips of the event "Birthday party" are clips with shot changes, whereas only 21% of successfully detected clips had shot changes. Shot changes include various effects such as cut, dissolve, or fade in/out. These effects, which have no direct relation to events, may be extracted as local features. Accordingly, such features may become noise in model training and degrade detection performance. We can avoid this problem by detecting shot changes beforehand using some shot boundary detection techniques and exclude features related to them. Other missed detections may be due to the variety of objects or backgrounds in the events. For instance, "Grooming an animal" includes animals such as dogs, cats, horses, birds, and snakes. This variety makes it difficult to learn the event model. Unsupervised clustering of keyframes is a promising way to solve this problem.

5. CONCLUSION

We devised a general method for multimedia event detection using GMM supervectors and SVMs. It performed better (mean MNDC 0.534) than the previous studies using Bag-of-Words, semantic event models, and Earth Mover's Distance (mean MNDC 0.565). However, there is still a lot of room for improvement. For example, devising new features and shot boundary detection are promising ways to improve performance.

6. REFERENCES

[1] C. Xu, J. Wang, K. Wan, Y. Li, and L. Duan, "Live sports event detection based on broadcast video and web-casting text," in *Proc. ACM Multimedia*, 2006, pp. 221–230.

[2] L. Ying, S. Narayanan, and C.C.J. Kuo, "Content-based movie analysis and indexing based on audiovisual cues," *IEEE Trans. CSVT*, vol. 14.

[3] M. Cristani, M. Bicego, and V. Murino, "Audio-Visual Event Recognition in Surveillance Video Sequences," *IEEE Trans. Multimedia*, vol. 9, pp. 257–267, 2007.

[4] K. Yan, R. Sukthankar, and M. Hebert, "Efficient visual event detection using volumetric features," in *Proc. ICCV*, 2005, vol. 1, pp. 166–173.

[5] J. Niebles, H. Wang, and L. Fei-Fei, "Unsupervised Learning of Human Action Categories Using Spatial-Temporal Words," *IJCV*, vol. 79, pp. 299–318, 2008.

[6] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez, and T. Tuytelaars, "A Thousand Words in a Scene," *IEEE Trans. PAMI*, vol. 29, no. 9, pp. 1575–1589, 2007.

[7] Y.-G. Jiang, X. Zeng, G. Ye, S. Bhattacharya, D. Ellis, M. Shah, and S.-F. Chang, "Columbia-UCF TRECVID2010 Multimedia Event Detection: Combining Multiple Modalities, Contextual Concepts, and Temporal Matching," in *Proc. TRECVID Workshop*, 2010.

[8] W. M. Campbell, D. E. Sturim, D. A. Reynolds, and A. Solomonoff, "SVM Based Speaker Verification using a GMM Supervector Kernel and NAP Variability Compensation," in *Proc. ICASSP*, 2006, vol. 1, pp. 97–100.

[9] N. Inoue and K. Shinoda, "A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems," in *Proc. ACM Multimedia*, 2011, pp. 1357–1360.

[10] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *IJCV*, vol. 60, pp. 91–110, 2004.

[11] X. Zhou, X. Zhuang, S. Yan, S. F. Chang, M. Hasegawa-Johnson, and T. S. Huang, "SIFT-Bag kernel for video event analysis," in *Proc. ACM Multimedia*, 2008, pp. 229–238.

[12] I. Laptev, "On Space-Time Interest Points," *IJCV*, vol. 64, pp. 107–123, 2005.

[13] N. Dalal, B. Triggs, and C. Schmid, "Human Detection Using Oriented Histograms of Flow and Appearance," in *Proc. ECCV*, 2006, pp. 428–441.

[14] J.-L. Gauvain and L. Chin-Hui, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Trans. Speech Audio Process*, vol. 2, pp. 291–298, 1994.

[15] The National Institute of Standards and Technology (NIST), "TREC video retrieval evaluation," http://www-nlpir.nist.gov/projects/trecvid/.