論文 / 著書情報
Article / Book Information

Title	Reusing Speech Techniques for Video Semantic Indexing
Author	Koichi Shinoda, Nakamasa Inoue
Journal/Book name	IEEE signal processing magazine, Vol. 30, No. 2, pp. 118-122
Issue date	2013, 3
DOI	http://dx.doi.org/10.1109/MSP.2012.2230520
URL	http://www.ieee.org/index.html
Copyright	(c)2013 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.

Reusing speech techniques for video semantic indexing

Koichi Shinoda and Nakamasa Inoue

Many techniques developed in speech research have been successfully employed in other fields, such as for example automatic video semantic indexing. In this application, a user submits a textual input query for an desired object or a scene to a search system, which returns video shots that include the object or scene. Recently, a new method using Gaussian-mixture-model (GMM) supervectors and support vector machines (SVM) was proven to be very effective. In this method, speech technology such as speaker verification and adaptation techniques play very important roles.

WHAT IS VIDEO SEMANTIC INDEXING?

An explosion of consumer video clips are now available on the Internet, and as a result, video search techniques are needed to make relevant clips easily accessible. Since most video clips provide poor text information about their contents, content-based video retrieval (CBVR) using pattern recognition techniques has been extensively studied. Most video clips are created by amateurs and thus their quality is usually low. Moreover, the objects to be searched may belong to very different semantic categories and therefore CBVR for consumer video is a very challenging task.

The US National Institute of Standards and Technology (NIST) has held the TREC Video Retrieval Evaluation (TRECVID) workshop every year since 2001 to promote CBVR research and development [1]. Many research organizations participate in this workshop, and compete with each other on their performance in several CBVR tasks. Their methods and results are open to the public on the TRECVID web page [2], which is a showcase of the state-of-the-art CBVR technologies.

An important task in TRECVID is video Semantic INdexing (SIN). This task has been conducted since 2002 and has had the largest number of participants amongst the various TRECVID tasks. In this task, a query comes in the form of a word or a phrase that is called a *concept* such as “night scape” or “dancing”. A search system should find *shots* including the concept from a large archive of Internet consumer video clips. In some previous studies for broadcast news or sports video, speech recognition was extensively used to obtain concepts (e.g., [3]). In this task for consumer videos, however, the focus is more on visual cues, since their transcribed speech is not accurate nor useful.

Most SIN methods have been based on the Bag of Visual Words (BoW) framework. In this framework, image features extracted from the video frames are clustered to form a codebook. For each shot, a code histogram is obtained by counting the number of occurrences of each code. This code histogram, a vector with a dimension equal to the codebook size, is expected to represent the characteristics of the shot. Support vector machines (SVM) are often used to detect the shots which include the target concept.

Video features used in this BoW framework are mostly image features developed for generic object recognition in still images. One of the most famous features is scale invariant feature transform

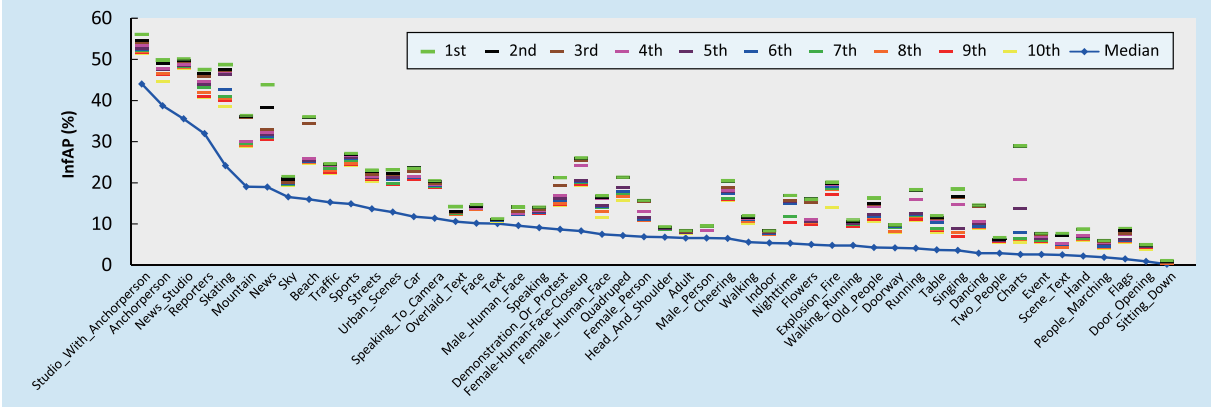

Figure 1: The result of 2011 SIN task for 50 features. InfAP is *inferred* AP which is estimated from sampling evaluation.

(SIFT) [4]. In SIFT, interest points are first detected by derivative operators applied to image pixels. The number of interest points in an image frame ranges from 10^3 to 10^4. Then, from the region around each point, features that are stable under deformations such as scale changes, rotations, and illumination changes are extracted. Their dimension is typically 128, but is often further reduced by principal component analysis (PCA). Many other related features have also been developed such as Color-SIFT, Histogram of Oriented Gradients (HOG), and Speeded Up Robust Feature (SURF).

The size of the SIN task is rapidly increasing year by year. In 2011, the total length of video data was 600 hours, where 264,673 shots were provided for developing the systems and 137,327 shots without any annotation were used for system evaluation. The number of concepts to be detected was 346. Each shot in the development data was annotated with concept labels. Here, a concept is involved in a shot when it appears in at least one frame of the shot and one shot can involve more than one concept. For each concept, each team submitted at most 2,000 shots ranked by their confidence. In 2011, 28 teams submitted their results.

The performance of a system for a concept is measured by average precision (AP), which is defined as:

$$\mathrm{AP} = \frac{1}{K}\sum_{k=1}^{K}\frac{p(k)\delta(k)}{k} \tag{1}$$

where K is the number of shots in which the system claims to detect the concept ($K \leq 2{,}000$), $p(k)$ is the number of shots actually involving the concept in the top-k shots in their submission, and $\delta(k)$ is an indicator function equaling 1 if the k-th shot is correct, and zero otherwise. Figure 1 shows APs for each concept in the 2011 SIN task. The detection performance varied largely among concepts, because the difficulty of detecting each concept and the number of shots for each concept differed greatly from concept to concept. If their corresponding amount of development data is small, the detector constructed may have poor performance. The performance of each search system is measured by Mean AP, which is AP averaged over all the concepts. The best Mean AP among all the 28 teams was 17.1%. While this value seems to be rather low, the top-10 results for most concepts are mostly correct (see Figure 2). This level of performance may be sufficient for most SIN search needs.

As described in the previous section, most techniques used in the SIN task have been imported from studies of recognizing objects in still images. Recently, however, many techniques have been proposed, which effectively utilize the characteristics of the video data. Previously, only features ex-

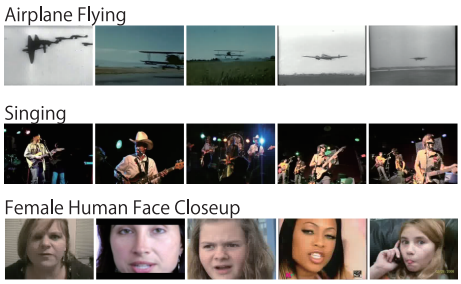

Figure 2: Examples of SIN results for three concepts.

tracted from a key frame of a shot were used, mainly due to the insufficiency of computation resources. Nowadays, features from many frames are often used, and contribute to increase the robustness of the detectors against various dynamic changes within a shot. Most Internet video clips provide not only video information but also audio information may enhance the SIN performance. For example, Mel-frequency cepstral coefficients (MFCC) significantly improved the detection performance for many concepts related to audio, such as “Infant”, “Car race”. When multiple features are used, one SVM is provided for each feature, and the outputs from those SVMs are combined to obtain the detection score.

APPROACHES USING SPEECH TECHNOLOGY

Important issues to be addressed are the low quality of the data, their large variety and the variable quality of the semantic labels. A larger model with more features is required, but the data insufficiency often deteriorates its performance. A system for a specific concept in a specific condition cannot be applied anymore because of the number of concepts increases every year and the data size doubles every year. A generic system that is robust against various changes such as quality and data size is clearly desirable.

The current situation of video semantic indexing reminds us of speech/speaker recognition in the 90’s. At that time, speech researchers were faced with a very similar problem. The solution they found was a robust data-driven approach which heavily relied on probability theory. Thanks to the advancement of computation technology, the same approach can now be readily used for video semantic indexing which requires much more computational resources than the speech tasks of 20 years ago.

In the two following sections, we introduce a video semantic indexing method [5] as an example of such approaches heavily using speech/speaker recognition technologies and that is an extension of the BoW framework to the probabilistic framework. This method obtained the highest Mean AP in the 2011 SIN task. Figure 3 illustrates its outline.

GAUSSIAN MIXTURE MODELS

In this method, a GMM is provided for each shot. Let $\chi = \{x \in \mathrm{R}^D\}$ be a set of input feature vectors with dimension D, which are extracted from a shot. This D is typically 32 for SIFT features after

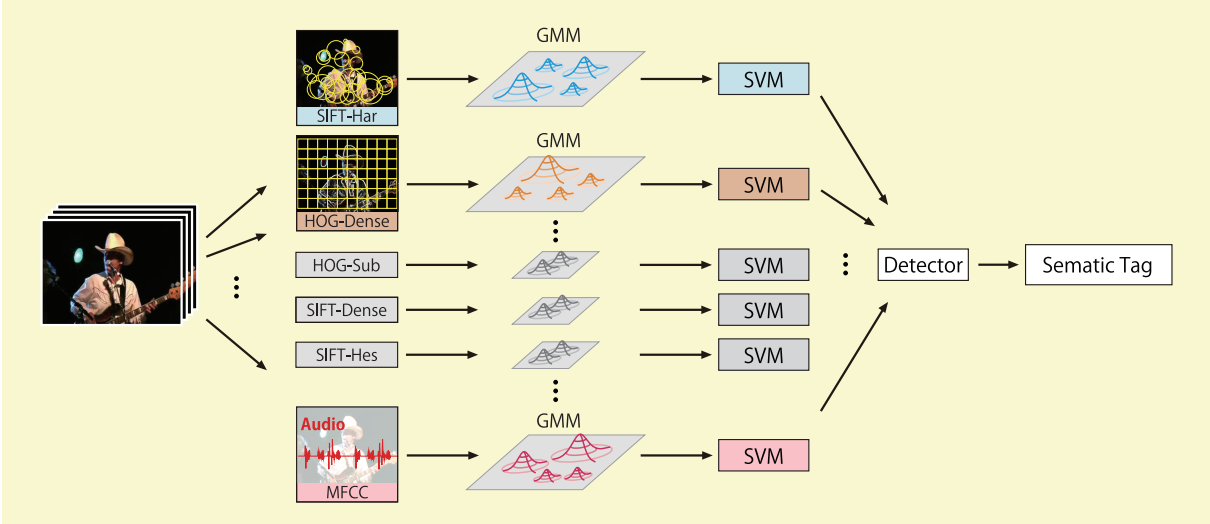

Figure 3: A video semantic indexing method [5] using six different features, each with one associated SVM.

PCA. Then the probability density function (pdf) of a shot GMM for x is:

$$g(x) = \sum_{k=1}^{K} w_k \mathcal{N}(x|\mu_k, \Sigma_k), \tag{2}$$

where w_k is the weight coefficient of the k-th mixture component which is a multivariate Gaussian pdf with mean μ_k and covariance matrix Σ_k. The number of mixture components, K, whose typical value ranges from 10^2 to 10^3, is usually set to be the same for all shots. The parameters μ_k, Σ_k, and w_k, are estimated from the feature vectors in χ using the Expectation-Maximization (EM) algorithm.

Here a mixture component k corresponds to a code in BoW, its mean vector μ_k corresponds to a code vector of the code, and its weight coefficient w_k can be regarded as the normalized occurrence count of the code in a code histogram. Like BoW, the spatial and temporal location of a feature is not explicitly used in a shot GMM. While only one code is assigned for one input in the BoW framework, one input belongs to many pdfs with different weights in a shot GMM. This *soft assignment* mitigates the effect from quantization errors, and hence brings robustness against data sparseness.

In the BoW approach, we count the number of occurrences of each code in a shot to construct its code histogram. The number of free parameters to be estimated is equal to the codebook size, whose typical value is around 10^3. For a shot GMM, we estimate the three kinds of parameters, w_k, μ_k and Σ_k. The number of free parameters is $K + K \times D + K \times D \times D$, which is much larger than that in BoW. While a shot GMM can contain much richer information than a code histogram, its parameters may not be precisely estimated by the small amount of data in a shot. We need to solve this data sparseness problem.

First, a feature vector of video features such as SIFT and MFCC is often made by transforming more primitive features such that different elements in a vector are less dependent on each other. We therefore ignore the off-diagonal elements in a covariance matrix. That is, we assume a covariance matrix is diagonal. Next, we further assume the same weight coefficients and the same diagonal covariance matrix are shared among all shots. That is, we estimate one GMM using all the shots in the training data and use its weight coefficients and covariance matrices for every shot. This GMM,

estimated from all the shots in the training data, is called a universal background model (UBM). This term was borrowed from *speaker verification* research.

SPEAKER ADAPTATION

The remaining problem is how to estimate mean vectors in a shot GMM. We need a method which robustly and precisely estimates them with a small amount of data available. Here we can use an approach from the speech field, developed from exactly the same motivation – *speaker adaptation*. Among the various speaker adaptation techniques, we use maximum *a posteriori* (MAP) adaptation [7].

In MAP adaptation, assuming a prior distribution of each mean vector μ_k of a shot GMM, a new mean vector $\hat{\mu}_k$ is estimated as the mode of its posterior probability,

$$\begin{aligned} \hat{\mu}_k &= \arg\max_{\mu_k} p(\mu_k|\chi) \\ &\propto \arg\max_{\mu_k} p(\chi|\mu_k)p(\mu_k). \end{aligned} \tag{3}$$

Here, for the prior distribution of mean vector μ_k, we choose a Gaussian distribution whose mean vector is that of the k-th mixture component in UBM, $\mu_k^{(\mathrm{U})}$. By solving Eq. (3), we obtain $\hat{\mu}_k$ as:

$$\begin{aligned} &\hat{\mu}_k = \frac{\tau\hat{\mu}_k^{(\mathrm{U})} + \sum_{i=1}^{n} c_{ik}x_i}{\tau + C_k}, \;\; c_{ik} = \frac{w_k^{(\mathrm{U})} g_k(x_i)}{\sum_{k=1}^{K} w_k^{(\mathrm{U})} g_k(x_i)}, \\ &C_k = \sum_{i=1}^{n} c_{ik}, \;\; g_k(x) = \mathcal{N}(x|\hat{\mu}_k^{(\mathrm{U})}, \hat{\Sigma}_k^{(\mathrm{U})}). \end{aligned} \tag{4}$$

Here, the parameters with superscript (U) are that of UBM. A hyper parameter τ controls the dependency of the estimate on the prior distribution, and c_{ik} is *a posteriori* probability of x_i being generated from the k-th mixture component. The sum C_k of c_{ik} over all the features x_i in the shot is called the occupancy count of the k-th mixture component, which corresponds to the number of inputs generated from the component.

As shown in Eq. (4), the MAP estimate $\hat{\mu}_k$ of the mean vector, which is obtained by MAP adaptation, is a weighted sum of UBM mean vector and its maximum-likelihood (ML) estimate. The weight between these two are controlled by the hyper parameter τ and the occupancy count C_k. As τ becomes larger, $\hat{\mu}_k$ gets closer to the UBM mean vector $\hat{\mu}_k^{(\mathrm{U})}$. When C_k is zero, $\hat{\mu}_k$ is identical to the UBM mean vector. As C_k becomes larger, $\hat{\mu}_k$ gets closer to the ML estimate. It is well known that ML estimation often fails to give good parameters when the amount of training data is small. On the contrary, MAP adaptation enables us to estimate the parameters more robustly in such a case by utilizing the UBM parameters as the prior information.

GMM SUPERVECTOR AND SVM

Previously, in GMM-based speaker verification, a likelihood ratio between the target GMM and the UBM was often used for classification. Recently, however, kernel methods such as SVMs have been proven to be more effective, especially when the amount of data for training each model is small. Here we apply one such method [8], which uses GMM supervector as the input for a SVM, to video semantic indexing.

First let us define the distance between two GMMs, which is needed to apply kernel methods. The distance between probabilistic models is naturally defined by Kullback-Leibler divergence (KLD) as:

$$D(g^a||g^b) = \int g^a(x)\log(g^a(x)/g^b(x))dx. \tag{5}$$

Since this KLD does not satisfy the Mercer's condition, it is difficult to directly apply it to kernel methods. Instead, we use its upper bound $d(g^a,g^b)$ given by the following log-sum inequality:

$$D(g^a||g^b) \leq d(g^a,g^b), \tag{6}$$

where

$$d(g^a,g^b) = \sum_{k=1}^{K} w_k(D(\mathcal{N}(\,\cdot\,;\mu_k^a,\Sigma_k)||(\mathcal{N}(\,\cdot\,;\mu_k^b,\Sigma_k))). \tag{7}$$

Here μ_k^a and μ_k^b are the mean vectors of the k-th component of GMM g^a and g^b respectively. Since the covariance matrix is diagonal, $d(g^a,g^b)$ is further simplified as:

$$d(g^a,g^b) = \frac{1}{2}\sum_{k=1}^{K} w_k(\mu_k^a-\mu_k^b)^{\top}\Sigma_k^{-1}(\mu_k^a-\mu_k^b), \tag{8}$$

which is a weighted sum of the squares of Mahalanobis distances.

Then, we define a GMM supervector $\phi(g)$ for GMM g as:

$$\phi(g) = \begin{pmatrix} \tilde{\mu}_1 \\ \tilde{\mu}_2 \\ \vdots \\ \tilde{\mu}_K \end{pmatrix}, \quad \tilde{\mu}_k = \sqrt{w_k^{(\mathrm{U})}}(\Sigma_k^{(\mathrm{U})})^{-\frac{1}{2}}\mu_k \tag{9}$$

This $\phi(g)$ is made by concatenating the mean vectors of all mixture components, each of which is weighted by its corresponding variance and the weight coefficient. Then, Eq. (8) is simplified as:

$$d(g^a,g^b) = \frac{1}{2}||\phi(g^a)-\phi(g^b)||^2. \tag{10}$$

Here we use the following radius basis function (RBF) kernel.

$$d(g^a,g^b) = \exp\left(-\gamma\|\phi(g^a)-\phi(g^b)\|^2\right), \quad \gamma = \frac{1}{\tilde{d}}, \tag{11}$$

where $\tilde{d}$ is the average of the distance between GMM supervectors over all shot GMMs.

A Fisher kernel is often used in classification with generative probabilistic models. A Fisher kernel for a GMM is defined for the 0th order statistics (weight coefficients), the 1st order statistics (mean vectors), and the 2st order statistics (covariances). The kernel method using GMM supervectors corresponds to a method using a Fisher kernel only for the 1st order statistics.

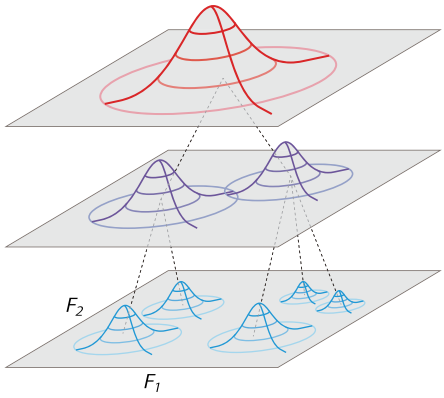

Figure 4: A tree-structured GMM

FAST ADAPTATION USING TREE-STRUCTURED GMMS

The calculation of the posterior probability c_{ik} in Eq. (4) is the most computationally expensive step in this video semantic indexing method. An approximation using a tree-structured GMM is introduced here to make this process faster [5].

A tree-structured shot GMM is defined by a set of nodes V, a set of edges E, and a set of Gaussian pdfs $G_{\text{TREE}} = \{g^{(v)} | v \in V\}$ as follows:

$$\mathscr{T} = (V, E, G_{\text{TREE}}). \tag{12}$$

Here each Gaussian pdf $g^{(v)}$ is determined to suffice the following two requirements:

(a) Each leaf node corresponds to a mixture component.

(b) A Gaussian pdf of each non-leaf node approximates the mixture pdfs of its child nodes.

A schematic view is shown in Figure 4.

A tree-structured GMM $\mathscr{T}$ is first constructed from UBM. Then for each input feature x_i of each input shot, the posterior probability is calculated starting from the root and progressing towards the leaves. If the probability goes below a threshold, the probability at a node is used for all mixture components it governs. By this method, we can largely reduce the computational costs required to calculate the posterior c_{ik} for every mixture k.

FEATURE DIRECTION

A consumer video can be regarded as a communication tool in which a message is encoded by a sender and is decoded by a receiver. A *universal grammar* exists for human to methodically decipher and interpret observed media data, whether it is audio or visual. Hence it is not far-fetched to reuse semantic analysis tools designed for audio data for video data as well. In such a sense, speech/speaker recognition technology and video semantic indexing can share the same methodology. This may be the reason why the techniques developed in speech area are often effective in video semantic indexing as shown in this article.

Video semantic indexing is still in the early developmental stage; the TRECVID SIN task corresponds to isolated word recognition with a limited vocabulary size in speech research. Various attempts to obtain higher-level semantics have recently started (e.g., [9]). TRECVID also defined a new task recently in 2010, which aims to extract a video clip including an *event*, such as "Getting a vehicle unstuck" from an Internet video archive. This direction again reminds speech researchers of their past history developing large vocabulary continuous speech recognition. Speech researchers can contribute a lot also in video processing as discussed in this column.

AUTHORS

Koichi Shinoda (shinoda@cs.titech.ac.jp) is an associate professor at the Dept. Computer Science, Tokyo Institute of Technology, Japan.

Nakamasa Inoue (inoue@ks.cs.titech.ac.jp) is a Ph.D candidate at the Dept. Computer Science, Tokyo Institute of Technology, Japan.

References

[1] A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation Campaigns and TRECVid," *Proc. ACM Multimedia MIR workshop*, pp. 321-330, 2006.

[2] TREC Video Retrieval Evaluation. http://trecvid.nist.gov/

[3] A. G. Hauptmann and D. Lee, "Topic labeling of broadcast news stories in the informedia digital video library," *Proc. ACM Digital Libraries*, pp. 287-288, 1988.

[4] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", International Journal of Computer Vision, vol. 60, no. 2, pp. 91-110, 2004.

[5] N. Inoue, and K. Shinoda, "A fast and accurate video semantic-indexing system using fast MAP adaptation and GMM supervectors," IEEE Trans. Multimedia, vol. 14, No. 4, 2012.

[6] D. A. Reynolds, "Comparison of background normalization methods for text-independent speaker verification," Proc. Eurospeech, pp. 963-967, 1997.

[7] J. L. Gauvain and C.-H Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," IEEE Trans. Speech and Audio Processing, vol. 2, no. 2, pp. 291-298, 1994.

[8] W. M. Campbell and D. A. Raynolds, "Support vector machines using GMM supervectors for speaker verification," IEEE Signal Processing Letters, vol. 13, no. 5, pp. 308-311, 2006.

[9] M. Naphade, S.-F. Chang, A. Hauptmann and J. Curtis, "Large-scale Concept Ontology for Multimedia," *IEEE Multimedia*, vol. 13, no. 3, pp. 86-91, 2006.