Correlative Multilabel Video Annotation with Temporal Kernels

GUO-JUN QI
University of Science and Technology of China
XIAN-SHENG HUA and YONG RUI
Microsoft Corporation
JINHUI TANG
University of Science and Technology of China
TAO MEI
Microsoft Corporation
MENG WANG
University of Science and Technology of China
HONG-JIANG ZHANG
Microsoft Corporation

Automatic video annotation is an important ingredient for semantic-level video browsing, search and navigation. Much attention has been paid to this topic in recent years. These researches have evolved through two paradigms. In the first paradigm, each concept is individually annotated by a pre-trained binary classifier. However, this method ignores the rich information between the video concepts and only achieves limited success. Evolved from the first paradigm, the methods in the second paradigm add an extra step on the top of the first individual classifiers to fuse the multiple detections of the concepts. However, the performance of these methods can be degraded by the error propagation incurred in the first step to the second fusion one. In this article, another paradigm of the video annotation method is proposed to address these problems. It simultaneously annotates the concepts as well as model correlations between them in one step by the proposed *Correlative Multilabel* (CML) method, which benefits from the compensation of complementary information between different labels. Furthermore, since the video clips are composed by temporally ordered frame sequences, we extend the proposed method to exploit the rich temporal information in the videos. Specifically, a temporal-kernel is incorporated into the CML method based on the discriminative information between *Hidden Markov Models* (HMMs) that are learned from the videos. We compare the performance between the proposed approach and the state-of-the-art approaches in the first and second paradigms on the widely used TRECVID data set. As to be shown, superior performance of the proposed method is gained.

Categories and Subject Descriptors: H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*Indexing methods*; I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding—*Video analysis*

General Terms: Algorithms, Theory, Experimentation

Additional Key Words and Phrases: Video annotation, multilabeling, concept correlation, temporal kernel

ACM Reference Format:

Qi, G.-J., Tang, J., Wang, M., Hua, X.-S., Rui, Y., Mei, T., and Zhang, H.-J. 2008. Correlative multilabel video annotation with temporal kernels. ACM Trans. Multimedia Comput. Commun. Appl. 5, 1, Article 3 (October 2008), 27 pages. DOI = 10.1145/1404880.1404883 http://doi.acm.org/10.1145/1404880.1404883

Authors' addresses G.-J. Qi, Department of Automation, University of Science and Technology of China, 96 Jinzhai Road, Hefei, Anhui 230027, China; email: qgj@mail.ustc.edu.cn; J. Tang, M. Wang, Department of Electronic Engineering and Information Science, University of Science and Technology of China, 96 Jinzhai Road, Hefei, Anhui 230027, China; email: {jhtang, wangmeng}@mail.ustc.edu.cn; X.-S. Hua, T. Mei, Y. Rui, H.-J. Zhang, Microsoft, 49 Zhichun Road, 100080 Beijing, China; email: {xshua, tmei, yongrui, hjzhang}@microsoft.com.

1. INTRODUCTION

With the explosive emergence of considerable videos on the Internet (e.g., Youtube, VideoEgg, Yahoo! Video, and many videos on personal home pages and blogs), effective indexing and searching these video corpus becomes more and more attractive to users. As a basic technique in video index and search, semantic-level video annotation (i.e., the semantic video concept detection) has been an important research topic in the multimedia research community [Naphade 2002; Snoek et al. 2006]. It aims at annotating videos with a set of concepts of interest, including scenes (e.g., urban, sky, mountain), objects (e.g., airplane, car, face), events (e.g., explosion-fire, people-marching) and certain named entities (e.g., person, place) [Naphade et al. 2005; Snoek et al. 2006]. Many efforts have been made on developing concept detection methods that can bridge the well known semantic "gap" between the low-level features and high-level semantic concepts [Hauptmann et al. 2007]. Among these efforts, some have paid their attentions on detecting specific concepts, such as object detection based on the bag-of-feature model [Jiang et al. 2007]. Recently, more efforts have been made on annotating video concepts in a generic fasion. For example, Naphade et al. [2006] build a large-scale concept ontology for generic video annotation and Snoek et al. [2006] construct an ontology of 101 concepts from News video as well. In order to annotate these generic video concepts, Yanagawa et al. [2007] build a set of baseline detectors for 374 LSCOM concepts Naphade et al. [2006] by using Support Vector Machine (SVM) and Wang et al. [2007] attempt to leverage diverse features to detect different video concepts. On the other hand, Snoek et al. [2006] propose a novel pathfinder to utilize the authoring information to help index the generic multimedia data.

In contrast to the above generic video annotation algorithms, in this paper we are concerned on a multilabel video annotation process where a video can be annotated by multiple labels at the same time. We attempt to explore the correlations between different labels to leverage them for improving the annotation performance on generic video concepts. These multilabeled videos commonly exist in many real-world video corpus, for example, most of the videos in the widely-used TRECVID dataset [Smeaton et al. 2006] are annotated by more than one label from a set of 39 different concepts. Figure 1 illustrates some keyframes of the videos associated with their multiple labels. For example, a video can be classified as "person," "walking_running," and "road" simultaneously. In contrast to the multilabel problem, multiclass annotation only assigns one concept to each video. In most real-world video annotations, such as TRECVID annotations and the users' tags on many video-sharing website, the videos are often multilabeled by a set of the concepts rather than only a single one. Since it involves nonexclusive classification of multiple concepts, multilabel annotation is much more complex than multiclass annotation. We will focus on multilabel video annotation in this paper. It is worth noting that the proposed algorithm is different from the knowledge based algorithm such as Koskela et al. [2007]. This knowledge based algorithm incorporates the prior knowledge of concept similarities to help the generic video annotation. In contrast we adopt a data-driven method in this article to explore the correlations between different labels. The details of the proposed algorithm will be presented later in this paper.

1.1 Video Annotation with Multiple Labels

Multilabel video annotation has evolved through two paradigms: individual concept detection and annotation, and *Context Based Conceptual Fusion* (CBCF) [Jiang et al. 2006] annotation. In this article, we propose the third paradigm: the unifying multilabel annotation. We next review these three paradigms.

1.1.1 *Paradigm I: Individual Concept Annotation.* The annotation methods in the first paradigm are individual concept detectors; that is, they annotate the video concepts individually and independently. They neglect the rich correlations between the video concepts. In more detail, these methods translate the multilabel annotations into some independent concept detectors that individually

Outdoor	T	T	T	T	T	T	T
Face	T	T	T	T	T	T	T
Person	T	T	T	T	T	T	T
People-Marching	F	F	F	T	T	F	F
Road	T	T	T	T	T	T	T
Walking_running	T	T	T	T	T	T	T

Fig. 1. Some multilabeled examples from TRECVID dataset. “T” and “F” mean the positive and negative labels for corresponding concepts respectively.

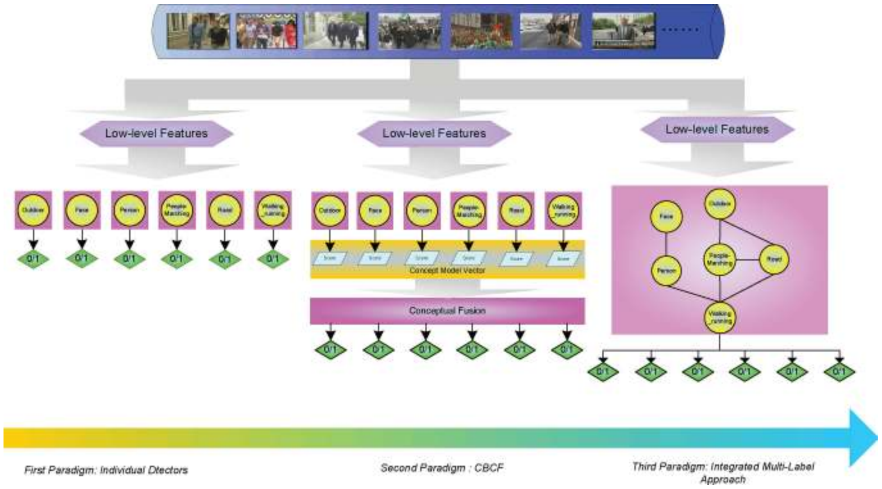

Fig. 2. The multilabel video annotation methods in three paradigms. From leftmost to the rightmost, they are the individual SVM, CBCF, and our proposed CML.

assign presence/absence labels into each sample. Most classical detectors can be categorized into this paradigm. For example, SVM [Cristianini and Shawe-Taylor 2000] with one-against-the-other strategy attempts to learn a set of detectors, each of which independently models the presence/absence of a certain concept. Other examples of this paradigm include Maximum Entropy Models (MEM) [Nigam et al. 1999], Manifold Ranking (MR) [Tang et al. 2007] etc. In Figure 2, we give an illustration of this paradigm in the leftmost flowchart. As depicted, a set of individual SVMs is learned for video concept annotation independently. In brief, the core of this paradigm is to formulate the video annotation as a collection of independent binary classifiers.

However in many real-world problems, video concepts do often exist correlatively with each other, rather than appearing in isolation. So the individual annotation only achieves limited success. For example, the presence of “Crowd” often occurs together with the presence of “People,” while “Boat_Ship” and “Truck” commonly do not co-occur. On the other hand, compared to simple concepts which can be directly modeled from low-level features, some complex concepts, for example, “People-Marching,” are really difficult to be individually modeled due to the semantic gap between these concepts and low-level features. Instead, these complex concepts can be better inferred based on the label correlations with the other concepts. For instance, the presence of "People-Marching" can be boosted if both "Crowd" and "Walking_Running" occur in a video. Therefore, it will be very helpful to exploit the label correlations when annotating the multiple concepts together.

1.1.2 *Paradigm II: Context-Based Conceptual Fusion Annotation.* As a step towards more advanced video annotation, the second paradigm is built atop the individual concept detectors. It attempts to refine the detection results of the binary concept detectors with a Context Based Concept Fusion strategy. Many algorithms can be categorized into this paradigm. For example, Wu et al. [2004] use an ontology-based multiclassification learning for video concept detection. Each concept is first independently modeled by a classifier, and then a predefined ontology hierarchy is investigated to improve the detection accuracy of the individual classifiers. Smith and Naphade [2003] present a two-step Discriminative Model Fusion approach to mine the unknown or indirect relationship between specific concepts by constructing model vectors based on detection scores of individual classifiers. A SVM is then trained to refine the detection results of the individual classifiers. The center flowchart of Figure 2 shows such a second-paradigm approach. Alternative fusion strategy can also be used; for example, Hauptmann et al. [2004] propose to use Logistic Regression to fuse the individual detections. Jiang et al. [2006] use a Context Based Concept Fusion-based learning method. Users are involved in their approach to annotate a few concepts for extra videos, and these manual annotations were then utilized to help infer and improve detections of other concepts. Naphade et al. [2002] propose a probabilistic Bayesian Multinet approach to explicitly model the relationship between the multiple concepts through a factor graph which is built upon the underlying video ontology semantics. Yan et al. [2006] mine the relationship between the detection results of different concepts by a set of various probabilistic graphical models. Zha et al. [2007] propose to leverage the pairwise concurrent relations between different labels to refine the video detection output by individual classifiers of the concepts.

Intuitively it is reasonable to leverage the context-based conceptual information to improve the accuracy of the concept detectors. However there also exist some experiments to show that the Context Based Concept Fusion methods do not have a consistent improvement over the individual detectors. Its overall performance can even be worse than the binary-based detectors. For example, in Hauptmann et al. [2004] at least 3 out of 8 concepts do not gain better performance by using the conceptual fusion with a linear regression classifier atop the uni-concept detectors. The unstable performance gain is due to the following reasons:

(1) Context Based Concept Fusion methods are built atop the independent binary detectors with a second step to fuse them. However, the output of the individual independent detectors can be unreliable and therefore their detection errors can propagate to the second fusion step. As a result, the final annotations can be corrupted by these incorrect prediction. From a philosophical point of view, the Context Based Concept Fusion methods do not follow the *principle of Least-Commitment* espoused by D. Marr [Marr 1982], because they are prematurely committed to irreversible individual predictions in the first step which can or cannot be corrected in the second fusion step.

(2) A secondary reason comes from the insufficient data for the conceptual fusion. In Context Based Concept Fusion methods, the samples need to be split into two parts for each step and the samples for the conceptual fusion step are usually insufficient compared to the samples used in the first training step. Unfortunately, the correlations between the concepts are usually complex, and insufficient data can lead to "over fitting" in the fusion step, thus the obtained prediction lacks the generalization ability.

1.1.3 *Paradigm III: Unifying Multilabel Annotation.* In this paper, we will propose the third paradigm of video annotation to address the problem faced in the first and second paradigms. This new paradigm will simultaneously model both the individual concepts and their correlations in a unifying formulation, and the *Principle of Least Commitment* will be obeyed. The rightmost flowchart of Figure 2 illustrates the proposed *Correlative Multilabel* (CML) method. As we can see, this method has the following advantages compared to the second Context Based Concept Fusion paradigm:

(1) The approach follows the *Principle of Least Commitment* [Marr 1982]. Because the learning and optimization is done in a single step for all the concepts simultaneously, it does not have the error propagation problem as in Context Based Concept Fusion.
(2) The entire samples are efficiently used simultaneously in modeling the individual concepts as well as their correlations. The risk of overfitting due to the insufficient samples used for modeling the conceptual correlations is therefore significantly reduced.

To summarize, the first paradigm does not address concept correlation. The second paradigm attempts to address it by introducing a separate second correlation step. In contrast, the third paradigm addresses the correlation issue at the root in a single optimization step. We will see that such a joint optimization model can be formulated as a convex optimization problem and thus a global optimum can be found.

1.2 Video Annotation with Temporal-Ordered Sequences

Besides the given multilabel problem, it is also an important issue to leverage the rich temporal information in the videos to boost the video annotation, especially for annotating the event-related concepts, such as "airplane-flying," "riot," "people-marching," etc.

There already exist some research works that attempt to utilize temporal information for video annotation. These researches have evolved through two research categories. In the first category, statistical models of feature dynamics are used to represent and detect video semantics. For example, Xie and Chang [2002] proposed to detect and segment the "play" and "break" events in soccer videos by learning the dynamics of the color and motion features with *Hidden Markov Model* (HMM). This method is only based on low-level feature dynamics to construct a generative model and ignores the other intuitive semantic components, such as visual concept interactions [Ebadollahi et al. 2006]. For example, while detecting "airplane-flying," it is helpful to detect whether "sky," "airplane" occurs.

The second research category detects the video events by exploiting the concept interactions. For example, Ebadollahi et al. [2006] propose to leverage stochastic temporal processes in the concept space to model the video events. This method aims at learning the dynamics of concurrent concepts from examplars of an event in a pure data-driven fashion. However, these concurrent concepts are obtained from the output of some prelearned concept detectors, which are often not robust enough to give reliable concept predictions. Therefore, the errors in the first step of concept predictions can propagate to the second step where we learn the concept dynamics of the video events. It also violates the principle of least commitment [Marr 1982] so that the errors incurred in the individual concept detector cannot be corrected in the second step of learning concept dynamics. The same problem incurs in [Wang et al. 2006] as well. It first pretrains a set of mid-level keyword detectors, based on which a Conditional Random Fields (CRF) [Lafferty et al. 2001] [Kumar and Hebert 2003] are used to capture the interactions between the noisy predictions of these keyword detectors.

To address the above problem, we will introduce a temporal kernel under the proposed correlative multilabel formulation. It can leverage the concept interactions as well as low-level feature dynamics to boost the video event detections. Specifically, it constructs a temporal kernel by revealing the discriminative information between the statistical models that are learned from the videos. As will be seen later, it avoids the two-step method in which the noisy outputs of the individual concept detector are propagated into the second conceptual dynamics. Instead, the concept interactions and low-level feature dynamics are captured in a unifying framework; thus the principle of least commitment is obeyed.

Furthermore, the proposed temporal kernel can be naturally incorporated to the proposed multilabel kernel without increasing the complexity of the algorithm.

The rest of the article is organized as follows. In Section 2, we give a detailed description of the proposed *Correlative Multilabel* (CML) method, including the classification model, the learning strategy. Furthermore we will explore the connection between the proposed approach and *Gibbs Random Fields* (GRFs) [Winkler 1995], based on which we can show an intuitive interpretation on how the proposed approach captures the individual concepts as well as their correlations. Section 3 details the temporal kernel for video annotation. This kernel can be naturally incorporated into CML kernel to form a *Correlative Multilabel Temporal* (CMLT) Kernel, which captures the high-level concept interactions and low-level feature dynamics in a unifying kernel machine. In Section 4, we will report experiments on the benchmark TRECVID data and show that the proposed approach has superior performance over the state-of-the-art algorithms in both first and second paradigms. Finally, we will conclude in Section 5.

2. CORRELATIVE MULTILABEL VIDEO ANNOTATION

In this section, we will introduce the proposed correlative multilabeling (CML) model for video semantic annotation. In Section 2.1, we will present the mathematical formulation of the multilabeling classification function, and show that how this function captures the correlations between the individual concepts and low-level features, as well as the correlations between the different concepts. Then in Section 2.2, we will give a probabilistic interpretation of the CML model based on Gibbs random fields. Based on this statistical model, we give an efficient inference approach to derive the label predictions of CML model.

2.1 Multilabel Classification Function

Before we move further, we first define some notations. Let $\boldsymbol{x} = (x_1, x_2, \dots, x_D)^T \in \mathcal{X}$ denote the input pattern representing feature vectors extracted from video clips; Let $\boldsymbol{y} \in \mathcal{Y} = \{+1, -1\}^K$ denote the K dimensional concept label vector of an example, where each entry $y_i \in \{+1, -1\}$ of $\boldsymbol{y}$ indicates the membership of this example in the ith concept. $\mathcal{X}$ and $\mathcal{Y}$ represent the input feature space and label space of the dataset, respectively. The proposed algorithm aims at learning a linear discriminative function

$$F(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{w}) = \langle \boldsymbol{w}, \theta(\boldsymbol{x}, \boldsymbol{y}) \rangle, \tag{1}$$

where $\theta(\boldsymbol{x}, \boldsymbol{y})$ is a vector function mapping from $\mathcal{X} \times \mathcal{Y}$ to a new feature vector that encodes the models of individual concepts as well as their correlations together (to be detailed later); $\boldsymbol{w}$ is the linear combination weight vector. With such a discriminative function, for an input pattern $\boldsymbol{x}$, the label vector $\boldsymbol{y}^*$ can be predicted by maximizing over the argument $\boldsymbol{y}$ as

$$\boldsymbol{y}^* = arg \max_{\boldsymbol{y} \in \mathcal{Y}} F(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{w}). \tag{2}$$

As to be presented in the next section, such a discriminative function can be intuitively interpreted in the Gibbs random fields (GRFs) [Winkler 1995] framework with the defined feature vector $\theta(\boldsymbol{x}, \boldsymbol{y})$. $\theta(\boldsymbol{x}, \boldsymbol{y})$ is a high-dimensional feature vector, whose elements can be partitioned into two types as follows. And as to be shown later these two types of elements actually account for modeling of individual concepts and their interactions, respectively.

Type I. The elements for *individual* concept modeling:

$$\theta^l_{d,p}(\boldsymbol{x}, \boldsymbol{y}) = x_d \cdot \delta[\![y_p = l]\!], l \in \{+1, -1\}, 1 \le d \le D, 1 \le p \le K, \tag{3}$$

where $\delta[\![y_p = l]\!]$ is an indicator function that takes on value 1 if the prediction is true and 0 otherwise; D and K are the dimensions of low level feature vector space $\mathcal{X}$ and the number of the concepts respectively. These entries of $\theta(\boldsymbol{x}, \boldsymbol{y})$ serve to model the connection between the low level feature $\boldsymbol{x}$ and the labels $y_k (1 \leq k \leq K)$ of the concepts. They have the similar functionality as the traditional SVM which models the relations between the low-level features and high-level concepts.

However, as we have discussed, it is not enough for a multilabeling algorithm to only account for modeling the connections between the labels and low-level features without considering the semantic correlations of different concepts. Therefore, another element type of $\theta(\boldsymbol{x}, \boldsymbol{y})$ is required to investigate the correlations between the different concepts.

Type II. The elements for concept correlations:

$$\theta_{p,q}^{m,n}(\boldsymbol{x}, \boldsymbol{y}) = \delta[\![y_p = m]\!] \cdot \delta[\![y_q = n]\!] m, n \in \{+1, -1\}, 1 \leq p < q \leq K, \tag{4}$$

where the superscripts m and n are the binary labels (positive and negative label), and subscripts p and q are the concept indices. These elements serve to capture all the possible pairs of the label correlations. Note that, both positive and negative relations are captured by these elements. For example, the concept “building” and “urban” is a positive concept pair that often co-occurs while “explosion fire” and “waterscape waterfront” is negative concept pair that usually does not occur at the same time.

Note that we can model high-order correlations among these concepts as well, but it will require more training samples. As to be shown in the experiments of Section 4, such an order-2 model successfully trades off between the model complexity and concept correlation complexity, and achieves significant improvement in the concept detection performance.

By concatenating these two types of elements together, we can obtain the feature vector $\theta(\boldsymbol{x}, \boldsymbol{y})$. It is not difficult to see that the dimension of vector $\theta(\boldsymbol{x}, \boldsymbol{y})$ is $2KD + 4\binom{K}{2} = 2K(D + K - 1)$. When K and D are large, the dimension of $\theta(\boldsymbol{x}, \boldsymbol{y})$ will be extraordinarily high. For example, if $K = 39$ and $D = 200$, $\theta(\boldsymbol{x}, \boldsymbol{y})$ will have $18,564$ dimensions. Fortunately, this vector is *sparse* thanks to the indicator function $\delta[\![\cdot]\!]$ in Equations (3) and (4). This is a key factor in the mathematical formulation. As a result, the kernel function (i.e., the dot product) between the two vectors, $\theta(\boldsymbol{x}, \boldsymbol{y})$ and $\theta(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}})$, can be represented in a very compact form as

$$\langle \theta(\boldsymbol{x}, \boldsymbol{y}), \theta(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}) \rangle = \langle \boldsymbol{x}, \tilde{\boldsymbol{x}} \rangle \sum\nolimits_{1 \leq k \leq K} \delta[\![y_k = \tilde{y}_k]\!] + \sum\nolimits_{1 \leq p < q \leq K} \delta[\![y_p = \tilde{y}_p]\!] \delta[\![y_q = \tilde{y}_q]\!] \tag{5}$$

where $\langle \boldsymbol{x}, \tilde{\boldsymbol{x}} \rangle$ is the dot product over the low-level feature vector $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$. We call this kernel the *Correlative Multilabel* (CML) Kernel and the corresponding video annotation method *Correlative Multilabel Video Annotation* in this article. It is worth noting that, any other kernel function $K(\boldsymbol{x}, \tilde{\boldsymbol{x}})$ (such as Gaussian Kernel, Polynomial Kernel) can be substituted for $\langle \boldsymbol{x}, \tilde{\boldsymbol{x}} \rangle$ as in the conventional SVMs, and *nonlinear* discriminative functions can then be introduced with the use of these kernels. In Appendix A, we will present the learning procedure of this model. As to be described, the above compact kernel representation will be used explicitly in the learning procedure instead of the original feature vector $\theta(\boldsymbol{x}, \boldsymbol{y})$.

Before we move further, we illustrate an example in Figure 3 to help the readers to understand how the kernel in Equation (5) is constructed and the individual concepts and their correlations are modeled by Equations (3) and (4). Suppose we use a six-dimensional feature vector $\boldsymbol{x} = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6]^T$ to represent an image, and we consider five concepts, that is, person, road, beach, car, and tree. Then for the image in this figure, the label vector is $\boldsymbol{y} = [1, 1, -1, -1, 1]^T$. We first construct a new feature vector $\theta(\boldsymbol{x}, \boldsymbol{y})$ which can be divided to two types of elements as in Equations (3) and (4). In this example, we can find the first entry of Type I elements, that is, $\theta_{1,1}^{1}$ is equal to the first dimension of the original feature, that is, 0.1, because the label of the first concept is true. On the other hand, the first entry $\theta_{1,2}^{1,1}$

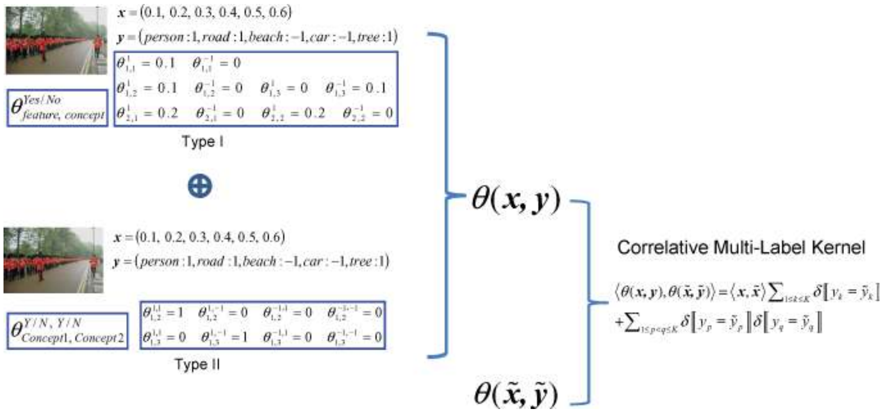
$x = (0.1, 0.2, 0.3, 0.4, 0.5, 0.6)$
$y = (person:1, road:1, beach:-1, car:-1, tree:1)$
$\theta^{Yes/No}_{feature,\ concept}$
$\theta^{1}_{1,1} = 0.1$ $\theta^{-1}_{1,1} = 0$
$\theta^{1}_{1,2} = 0.1$ $\theta^{-1}_{1,2} = 0$ $\theta^{1}_{1,3} = 0$ $\theta^{-1}_{1,3} = 0.1$
$\theta^{1}_{2,1} = 0.2$ $\theta^{-1}_{2,1} = 0$ $\theta^{1}_{2,2} = 0.2$ $\theta^{-1}_{2,2} = 0$
Type I
$\oplus$
$x = (0.1, 0.2, 0.3, 0.4, 0.5, 0.6)$
$y = (person:1, road:1, beach:-1, car:-1, tree:1)$
$\theta^{Y/N,\ Y/N}_{Concept1,\ Concept2}$
$\theta^{1,1}_{1,2} = 1$ $\theta^{1,-1}_{1,2} = 0$ $\theta^{-1,1}_{1,2} = 0$ $\theta^{-1,-1}_{1,2} = 0$
$\theta^{1,1}_{1,3} = 0$ $\theta^{1,-1}_{1,3} = 1$ $\theta^{-1,1}_{1,3} = 0$ $\theta^{-1,-1}_{1,3} = 0$
Type II
$\theta(x, y)$
$\theta(\tilde{x}, \tilde{y})$
Correlative Multi-Label Kernel
$\langle \theta(x,y), \theta(\tilde{x},\tilde{y}) \rangle = \langle x, \tilde{x} \rangle \sum_{1\le k\le K} \delta[\![y_k = \tilde{y}_k]\!]$
$+ \sum_{1\le p<q\le K} \delta[\![y_p = \tilde{y}_p]\!] \delta[\![y_q = \tilde{y}_q]\!]$

Fig. 3. An illustration of the correlative multilabel kernel. In this figure, we illustrate two types of feature elements for this correlative multilabel kernel. The type I features (top left capture the correlation between the low-level features and the high-level concepts, while the type II features capture the correlations between the different labels.

of Type II elements equals to 1 because the first two labels of this sample are all true. All the entries of $\theta(\boldsymbol{x}, \boldsymbol{y})$ can be computed by following the same rule. We illustrate all the entries of the obtained $\theta(\boldsymbol{x}, \boldsymbol{y})$ in the left part of this figure. After that, a new correlative multi-label kernel can be computed between two feature vectors $\theta(\boldsymbol{x}, \boldsymbol{y})$ and $\theta(\boldsymbol{x}, \boldsymbol{y})$ as in Equation (5), and a kernel machine can be trained accordingly as introduced in Appendix A.

2.2 A Justification—Gibbs Random Fields for Multi-Label Representation

In this section we give an intuitive interpretation of our multi-labeling model through Gibbs Random Fields (GRFs). Detailed mathematical introduction of GRFs can be found in [Winkler 1995]. We can rewrite Equation (1) as

$$F(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{w}) = \langle \boldsymbol{w}, \theta(\boldsymbol{x}, \boldsymbol{y}) \rangle = \sum\nolimits_{p \in \wp} D_p(y_p; \boldsymbol{x}) + \sum\nolimits_{(p,q) \in \mathcal{N}} V_{p,q}(y_p, y_q; \boldsymbol{x}) \tag{6}$$

and

$$\begin{aligned} D_p(y_p; \boldsymbol{x}) &= \sum\nolimits_{1 \le d \le D, l \in \{+1,-1\}} \boldsymbol{w}^{l}_{d,p} \theta^{l}_{d,p}(\boldsymbol{x}, \boldsymbol{y}) \\ V_{p,q}(y_p, y_q; \boldsymbol{x}) &= \sum\nolimits_{m,n \in \{+1,-1\}} \boldsymbol{w}^{m,n}_{p,q} \theta^{m,n}_{p,q}(\boldsymbol{x}, \boldsymbol{y}), \end{aligned} \tag{7}$$

where $\wp = \{i | 1 \le i \le K\}$ is a finite index set of the concepts with every $p \in \wp$ representing a video concept, and $\mathcal{N} = \{(p, q) | 1 \le p < q \le K\}$ is the set of interacting concept pairs. From the perspective of GRFs, $\wp$ is the set of sites of a random field and $\mathcal{N}$ consists of adjacent sites of the concepts. For example, in Figure 4, the corresponding GRF has 6 sites representing "Outdoor," "Face," "Person," "People-Marching," "Road," and "Walking_running," and these sites are interconnected by the concept interactions, such as (Outdoor, People-Marching), (Face, Person), (People-Marching, Walking_running), etc., which are included in the neighborhood set $\mathcal{N}$ of GRF. In the CML framework, the corresponding $\mathcal{N}$ consists of all pairs of the concepts, that is, this GRF has a fully connected structure.

Now we can define the energy function for GRF given an example $\boldsymbol{x}$ as

$$H(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w}) = -F(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{w}) = -\left\{ \sum\nolimits_{p \in \wp} D_p(y_p; \boldsymbol{x}) + \sum\nolimits_{(p,q) \in \mathcal{N}} V_{p,q}(y_p, y_q; \boldsymbol{x}) \right\}, \tag{8}$$

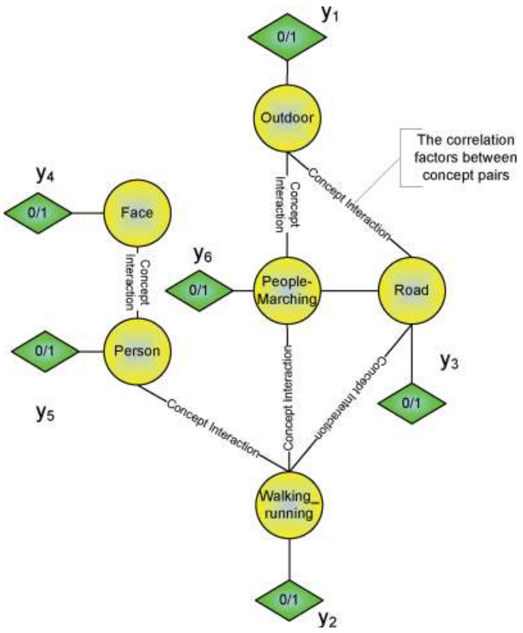

Fig. 4. Gibbs Random Fields for a correlative multi-label representation. The edges between concepts indicate the correlation factors $P_{p,q}(y_p, y_q|x)$ between concept pairs.

and thus we have the probability measure for a particular concept label vector $\boldsymbol{y}$ given $\boldsymbol{x}$ in the form

$$P(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w}) = \frac{1}{Z(\boldsymbol{x},\boldsymbol{w})}\exp\{-H(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w})\}, \tag{9}$$

where $Z(\boldsymbol{x},\boldsymbol{w}) = \sum_{\boldsymbol{y}\in\mathcal{Y}}\exp\left\{-H(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w})\right\}$ is the partition function. Such a probability function with an exponential form can express a wide range of probabilities that are strictly positive over the set $\mathcal{Y}$ [Winkler 1995]. It can be easily seen that when inferring the best label vector $\boldsymbol{y}$, maximizing $P(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w})$ according to the *Maximum A Posteriori* (MAP) criterion is equal to minimizing the energy function $H(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w})$ or equivalently maximizing $F(\boldsymbol{x},\boldsymbol{y};\boldsymbol{w})$, which is consistent with Equation (2). Therefore, the CML model can be connected to the above defined GRF.

Based on this GRF representation for multilabeling video concepts, the CML model now has a natural probability interpretation. Substitute Equation (8) into (9), we have

$$P(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w}) = \frac{1}{Z(\boldsymbol{x},\boldsymbol{w})}\prod\nolimits_{p\in\wp} P(y_p|\boldsymbol{x})\cdot\prod\nolimits_{(p,q)\in\mathcal{N}} P_{p,q}(y_p, y_q|\boldsymbol{x}), \tag{10}$$

where

$$P(y_p|\boldsymbol{x}) = \exp\{D_p(y_p;\boldsymbol{x})\}$$
$$P_{p,q}(y_p, y_q|\boldsymbol{x}) = \exp\{V_{p,q}(y_p, y_q;\boldsymbol{x})\}.$$

Here $P(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w})$ has been factored into two types of multipliers. The first type, that is, $P(y_p|\boldsymbol{x})$, accounts for the probability of a label y_p for the concept p given $\boldsymbol{x}$. These factors indeed model the relations between the concept label and the low-level feature $\boldsymbol{x}$. Note that $P(y_p|\boldsymbol{x})$ only consists of the first type of our constructed features in Equation (3), and thus it confirms our claim that the first type of the elements in $\theta(\boldsymbol{x},\boldsymbol{y})$ serves to capture the connections between $\boldsymbol{x}$ and the individual concept labels. The same discussion can be applied to the second type of the multipliers $P_{p,q}(y_p, y_q|\boldsymbol{x})$. These factors serve to model the correlations between the different concepts, and therefore our constructed features in Equation (4) account for the correlations of the concept labels.

The above discussion justifies the proposed model and the corresponding constructed feature vector $\theta(\boldsymbol{x}, \boldsymbol{y})$ for the multilabeling problem on video semantic annotation. In the following, we will give some further discussions based on this GRF representation.

2.2.1 *Concept Label Vector Prediction.* Once the classification function is obtained, the best predicted concept vector $\boldsymbol{y}^*$ can be obtained from Equation (2). The most direct approach is to enumerate all possible label vectors in $\mathcal{Y}$ to find the best one. However, the size of the set $\mathcal{Y}$ will become exponentially large with the increment of the concept number K, and thus the enumeration of all possible concept vectors is practically impossible. For example, when $K = 39$, the size is $2^{39} \approx 5.5 \times 10^{11}$.

Fortunately, from the revealed connection between CML and GRF in Section 2.2, the prediction of the best concept vector $\boldsymbol{y}^*$can be performed on the corresponding GRF form. Therefore, many popular approximate inference techniques on GRF can be adopted to predict $\boldsymbol{y}^*$, such as *Annealing Simulation*, G*ibbs Sampling*, etc. Specifically, these approximation techniques will be based on the output optimal dual variables $\alpha_i(\boldsymbol{y})$ in (40) of Appendix A. Following the above discussion about GRF representation, we can give the dual form of the GRF energy function accordingly. Such a dual energy function comes from Equation (40). Substituting (40) into (1) and considering the kernel representation (5), we can obtain the following equations:

$$\begin{aligned} F(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}; \boldsymbol{w}) &= \left\langle \sum\nolimits_{1 \le i \le n, \boldsymbol{y} \in \mathcal{Y}} \alpha_i(\boldsymbol{y}) \Delta\theta_i(\boldsymbol{y}), \theta(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}) \right\rangle \\ &= \sum\nolimits_{p \in \wp} \tilde{D}_p(\bar{y}_p; \bar{\boldsymbol{x}}) + \sum\nolimits_{(p,q) \in \mathcal{N}} \tilde{V}_{p,q}(\bar{y}_p, \bar{y}_q; \bar{\boldsymbol{x}}), \end{aligned} \tag{11}$$

where

$$\begin{aligned} \tilde{D}_p(\bar{y}_p; \bar{\boldsymbol{x}}) &= \sum\nolimits_{1 \le i \le n, \boldsymbol{y} \in \mathcal{Y}} \alpha_i(\boldsymbol{y}) k(\boldsymbol{x}_i, \bar{\boldsymbol{x}}) \{\delta \llbracket [y_{ip} = \bar{y}_p \rrbracket - \delta \llbracket [y_p = \bar{y}_p \rrbracket\} \\ \tilde{V}_{p,q}(\bar{y}_p, \bar{y}_q; \bar{\boldsymbol{x}}) &= \sum\nolimits_{1 \le i \le n, \boldsymbol{y} \in \mathcal{Y}} \alpha_i(\boldsymbol{y}) \{\delta \llbracket [y_i = \bar{y}_p \rrbracket \delta \llbracket [y_{iq} = \bar{y}_q \rrbracket - \delta \llbracket [y_p = \bar{y}_p \rrbracket \delta \llbracket [y_q = \bar{y}_q \rrbracket\}. \end{aligned} \tag{12}$$

And hence the dual energy function is

$$\tilde{H}(\bar{\boldsymbol{y}}|\bar{\boldsymbol{x}}, \boldsymbol{w}) = -\left\{ \sum\nolimits_{p \in \wp} \tilde{D}_p(\bar{y}_p; \bar{\boldsymbol{x}}) + \sum\nolimits_{(p,q) \in \mathcal{N}} \tilde{V}_{p,q}(\bar{y}_p, \bar{y}_q; \bar{\boldsymbol{x}}) \right\} \tag{13}$$

and the corresponding probability form of GRF can be written as

$$P(\bar{\boldsymbol{y}}|\bar{\boldsymbol{x}}, \boldsymbol{w}) = \frac{1}{\tilde{Z}(\bar{\boldsymbol{x}}, \boldsymbol{w})} \exp\{-\tilde{H}(\bar{\boldsymbol{y}}|\bar{\boldsymbol{x}}, \boldsymbol{w}), \} \tag{14}$$

where $\tilde{Z}(\bar{\boldsymbol{x}}, \boldsymbol{w}) = \sum_{\boldsymbol{y} \in \mathcal{Y}} \exp\{-\tilde{H}(\boldsymbol{y}|\bar{\boldsymbol{x}}, \boldsymbol{w})\}$ is the partition function of the dual energy function. With the above dual probabilistic GRF formulation, we use *Iterated Conditional Modes* (ICM) [Winkler 1995] for inference of $\boldsymbol{y}^*$ considering its effectiveness and easy implementation. Other efficient approximation inference techniques (e.g., *Annealing Simulation*, etc.) can also be directly adopted given the above dual forms.

2.2.2 *Concept Scoring.* The output of our algorithm given a sample $\boldsymbol{x}$ is the predicted binary concept label vector. However, for the video retrieval applications, we would like to give each concept of each sample a ranking score for indexing. With these scores, the retrieved video clips can be ranked according to the presence possibility of detecting the concept. Here we give a ranking scoring scheme based on

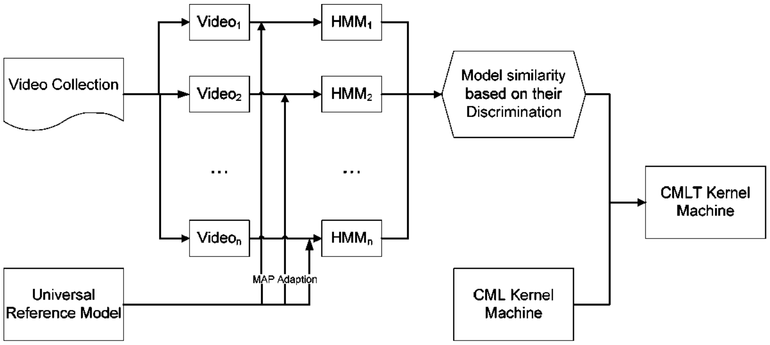

Fig. 5. The Correlative multilabel temporal kernel machine: It first adapts a universal reference model (URM) to a HMM for an individual video sequences. The model similarities can then be computed between these HMMs as temporal kernel based on their discrimination distances. By incorporating the temporal kernel into CML kernel machine, the CML temporal (CMLT) kernel machine can be obtained. Detailed algorithm is described in Section 3.

the probability form Equation (14). Given the predicted concept vector $\boldsymbol{y}^*$, the conditional expectation of y_p for the concept p can be computed as

$$E\left(y_p|\boldsymbol{x}, \boldsymbol{y}^*_{\wp\backslash p}\right) = P\left(y_p = +1|\boldsymbol{x}, \boldsymbol{y}^*_{\wp\backslash p}\right) - P\left(y_p = -1|x, \boldsymbol{y}^*_{\wp\backslash p}\right)$$

where

$$P\left(y_p|\boldsymbol{x}, \boldsymbol{y}^*_{\wp\backslash p}\right) = \frac{\exp\{-H(y_p \circ \boldsymbol{y}^*_{\wp\backslash p}|\boldsymbol{x}, \boldsymbol{w})\}}{Z_p} = \frac{\exp\{F(\boldsymbol{x}, y_p \circ \boldsymbol{y}^*_{\wp\backslash p}; \boldsymbol{w})\}}{Z_p} \tag{15}$$

and

$$Z_p\left(\boldsymbol{x}, \boldsymbol{y}^*_{\wp\backslash p}\right) = \sum\nolimits_{y_p \in \{+1,-1\}} \exp\left\{-H(y_p \circ \boldsymbol{y}^*_{\wp\backslash p}|\boldsymbol{x}, \boldsymbol{w})\right\} \tag{16}$$

is the partition function on the site p. The circle operator $\circ$ denotes the concatenation of two parts of labels into one. Then we can use this label expectation to rank the video clips for a certain concept.

3. CORRELATIVE MULTILABEL TEMPORAL KERNEL MACHINE FOR VIDEO ANNOTATION

In this section, we will introduce a temporal kernel machine under the above correlative multilabel video annotation framework. As mentioned in Section 1.2, the temporal information of videos is an important source to characterize the inherent video dynamics when annotating video concepts, especially for event concepts. To leverage this temporal information for video annotation, we will introduce a temporal kernel to represent the feature dynamics in this section.

3.1 A Temporal Kernel for Video Sequence

In CML Kernel (see Equation (5)), we have indicated that the dot product $\langle \boldsymbol{x}, \tilde{\boldsymbol{x}} \rangle$ over the low-level feature vector $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$ can be replaced by any other kernel function. Therefore, we design a temporal-based kernel that characterizes the dynamics of video sequences. To design such a temporal kernel, a distance measure $d(\boldsymbol{x}, \tilde{\boldsymbol{x}})$ between two videos $\boldsymbol{x}, \tilde{\boldsymbol{x}}$ can be first designed, and then a kernel can be computed through exponentiation as

$$K(\boldsymbol{x}, \tilde{\boldsymbol{x}}) = \exp\left\{-\frac{d(\boldsymbol{x}, \tilde{\boldsymbol{x}})}{\sigma^2}\right\} \tag{17}$$

where σ is the kernel radius. As is well known, the *Kullback-Leibler Divergence* (KLD) is a well-defined distance measure in information theory [Cover and Thomas 1991]. It can be used to compute the distribution distance between two statistical models. Therefore, if dynamic models are constructed to capture the temporal dynamics of video sequences, KLD can be computed between them. In this paper, we select Hidden Markov Models (HMMs) as such dynamic models. Specifically, for a video sequence, we denote its observations as $O = \{o_t, t = 1, \ldots, \mathcal{T}\}$ where each o_t as the feature vectors for frame t in the video. Let there be Q states $\{1, \ldots, Q\}$ and the state of each frame t is denoted by s_t. The transition probability $a_{i,j}$ denotes the state transition between the state i and j and $\pi = [\pi_1 \ \pi_2 \ \cdots \ \pi_Q]^T$ denotes the initial state distribution. For each state s_t, the observation o_t is generated according to an distribution $P(o_t|s_t)$. In this paper, we use *Gaussian Mixture Model* (GMM) as this observation distribution:

$$b_i(o_t) = P(o_t|s_t = i) = \sum\nolimits_{l=1}^{n} \lambda_l^i \mathrm{N}\big(o_t|\mu_l^i, \Sigma_l^i\big), \tag{18}$$

where $\lambda_l^q, \mu_l^q, \Sigma_l^q$ is the mixing coefficient, the mean vector and covariance matrix of lth Gaussian component respectively, given the current state is i. For simplicity, the covariance matrix is assumed to be diagonal.

Given two video sequences and their respective HMMs $\Theta, \tilde{\Theta}$ where Θ and $\tilde{\Theta}$ denote the underlying parameter spaces of two respective HMMs, We can compute the KLD [Cover and Thomas 1991] between them as

$$D_{KL}\big(\Theta||\tilde{\Theta}\big) = \int P(O|\Theta) \log \frac{P(O|\Theta)}{P(O|\tilde{\Theta})}. \tag{19}$$

However, there exists no closed form expression for the KLD between these two HMMs. The most straightforward approach to computing this KLD is to use the Monte Carlo simulation [Berg 2004], which requires high computational cost. In this section, we will introduce an alternative approximation approach that can be computationally more efficient than the Monte Carlo approach [Liu et al. 2007]. It aims at computing an upper bounded approximation of KLD between two HMMs as follows

$$\widehat{D}_{KL}(\Theta||\tilde{\Theta}) \leq \sum_{i=1}^{Q} \gamma_i \{D_{KL}(b_i||\tilde{b_i}) + D_{KL}(a_{i,\cdot}||\tilde{a}_{i,\cdot})\}, \tag{20}$$

where γ is the stationary distribution for the model Θ, that is, $\gamma^T A = \gamma^T, \lim_{t \to \infty} \pi^T A^t = \gamma^T,$ where $A = (a_{i,j})$ is the transition matrix of HMM Θ. b_i and $\tilde{b}_i$ are observation distribution defined in Equation (18) while $a_{i,\cdot}$ and $\tilde{a}_{i,\cdot}$ are the transition probability given the current state is i for the two HMMs Θ and $\tilde{\Theta}$. The detailed proof of the above upper bounds can be found in Appendix B.

Similarly, the upper bound of the reverse KLD rate is

$$\widehat{D}_{KL}(\tilde{\Theta}||\Theta) \leq \sum_{i=1}^{Q} \tilde{\gamma}_i \{D_{KL}(\tilde{b_i}||b_i) + D_{KL}(\tilde{a}_{i,\cdot}||a_{i,\cdot})\}, \tag{21}$$

where $\tilde{\gamma}$ is the stationary distribution of the model $\tilde{\Theta}$.

Thus with the above upper bounds of the KLD between HMMs, the symmetric KLD rate is

$$\begin{aligned} D(\Theta||\tilde{\Theta}) &= \frac{1}{2}\{\widehat{D}_{KL}(\Theta||\tilde{\Theta}) + \widehat{D}_{KL}(\tilde{\Theta}||\Theta)\} \\ &\leq \frac{1}{2}\sum_{i=1}^{Q}\gamma_i\{D_{KL}(b_i||\tilde{b}_i) + D_{KL}(a_{i,\cdot}||\tilde{a}_{i,\cdot})\} \\ &+ \frac{1}{2}\sum_{i=1}^{Q}\tilde{\gamma}_i\{D_{KL}(\tilde{b}||b_i) + D_{KL}(\tilde{a}_{i,\cdot}||a_{i,\cdot})\}. \end{aligned} \tag{22}$$

Substituting the above upper bound of the symmetric KLD rate into Equation (17), we can obtain the temporal kernel between two video sequences as

$$K(\Theta, \tilde{\Theta}) = \exp\left\{-\frac{\sum_{i=1}^{Q}\gamma_i\left\{D_{KL}(b_i||\widetilde{b_i})+D_{KL}(a_{i,\cdot}||\tilde{a}_{i,\cdot})\right\}+\sum_{i=1}^{Q}\tilde{\gamma}_i\left\{D_{KL}(\widetilde{b_i}||b_i)+D_{KL}(\tilde{a}_{i,\cdot}||a_{i,\cdot})\right\}}{2\sigma^2}\right\}. \tag{23}$$

Note that we use an upper bound to approximate the true KL distance between the two HMMs, thus their corresponding kernel according to the above equation may not be positive-definite. However, there are many solutions to address this problem. For example, Zhang et al. [2006] suggest computing the smallest eigenvalue of the kernel matrix, and if it is negative, its absolute value can be added to the diagonal of the kernel matrix. This method can be justified as follows. The kernel matrix can be explained intuitively as similarities between images. Adding a positive value to the diagonal only enhances "self-similarities" and it does not affect the similarities among images. Moreover, in practise, we have found the approximate upper bounded KLD distance gives a tight enough approximation to the true KLD so that the computed kernel usually satisfies the positive-definite condition. Thus the above technique is scarcely used.

With the above temporal kernel, we can define the Correlative Multilabel Temporal Kernel (CMLTK) by incorporating Equation (23) into Equation (5) as

$$\begin{aligned} K(\langle\theta(\boldsymbol{x}, \boldsymbol{y}), \theta(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}})\rangle) = \exp\left\{-\frac{\sum_{i=1}^{Q}\gamma_i\left\{D_{KL}(b_i||\widetilde{b_i})+D_{KL}(a_{i,\cdot}||\tilde{a}_{i,\cdot})\right\}+\sum_{i=1}^{Q}\tilde{\gamma}_i\left\{D_{KL}(\widetilde{b}||b_i)+D_{KL}(\tilde{a}_{i,\cdot}||a_{i,\cdot})\right\}}{2\sigma^2}\right\} \\ \cdot\sum\nolimits_{1\leq k\leq K}\delta[\![[y_k = \tilde{y}_k]\!] + \sum\nolimits_{1\leq p<q\leq K}\delta[\![[y_p = \tilde{y}_p]\!]\delta[\![[y_q = \tilde{y}_q]\!] \end{aligned} \tag{24}$$

Such a multilabel temporal kernel considers not only the concept interactions between each other but also the temporal evolution of video sequences. In this article, we call Equation (24) the *Correlative Multilabel Temporal* (CMLT) Kernel.

Finally, the KLD between the two GMMs distributions $b_i, \tilde{b}_i$ in these equations can be approximated through unscented transform [Goldberger and Aronowitz 2005]; that is,

$$D_{KL}(b_i||\tilde{b_i}) = \frac{1}{2d}\sum\nolimits_{l=1}^{n}\lambda_l^i\sum_{k=1}^{2d}\log\frac{\mathrm{N}(x_{l,k}^i|\mu_l^i, \Sigma_l^i)}{\mathrm{N}(x_{l,k}^i|\tilde{\mu}_l^i, \tilde{\Sigma}_l^i)}, \tag{25}$$

where d is the dimension of the observed feature vectors, and $x_{l,k}^i$ is the "sigma" points defined as

$$\begin{aligned} x_{l,k}^i &= \mu_l^i + \left(\sqrt{d\,\Sigma_l^i}\right)_k, \quad k = 1, \ldots, d \\ x_{l,d+k}^i &= \mu_l^i - \left(\sqrt{d\,\Sigma_l^i}\right)_k, \quad k = 1, \ldots, d. \end{aligned} \tag{26}$$

These sample points completely capture the true mean and variance of the Gaussian distribution $\mathrm{N}(x|\mu_l^i, \Sigma_l^i)$, that is, the l-th component distribution given its corresponding state is i.

3.2 A Universal Reference Model

As we have stated, we use an upper bound to approximate the intractable exact KLD between two HMMs. These two models have the same state number Q. However, since they are trained independently on their own video sequences, the correspondence between their respective underlying states may not be in the same order from 1 to Q. Such an inconsistency between the states in the two models can lead to an upper bound that is not tight enough. To obtain a tighter bound, we can first train a *Universal Reference Model* (URM) from referential sequences, for example, some video sequences from the training set. Then, given a new video, its HMM can be adapted from this URM. Since the models are all adapted from this URM, the states will have a reasonable correspondence between the models. Thus, the obtained upper bound will be much tighter than that computed from the independently trained models.

In this article, the standard *Maximum A Posteriori* (MAP) technique [Gauvain and Lee 1994] is used to adapt the HMM. Formally, given the parameters Θ^{URM} of the URM and an observation O of the new video sequence, we estimate the new HMM Θ. We use Θ^{URM} as the initial parameter. As suggested in Gauvain and Lee [1994], the standard *Expectation-Maximization* (EM) algorithm is then applied to update Θ repeatedly until convergence except for the mean vector of GMMs; that is,

$$\mu_l^i \leftarrow \alpha \cdot \mu_l^i + (1-\alpha) \cdot \frac{\sum_{t=1}^{\mathcal{T}} o_t \cdot P(s_t = i, m_t^i = l|O, \Theta)}{\sum_{t=1}^{\mathcal{T}} P(s_t = i, m_t^i = l|O, \Theta)}, \tag{27}$$

where m_t^i indicates the mixture component given the state is i at time slice t and α is the weighting factor giving the bias between the previous estimate and the current one. Following the suggestion in Gauvain and Lee [1994], we will set α to be 0.7 in the experiment. The update rules for all the other parameters follow the EM algorithm.

4. EXPERIMENTS

In this section, we conduct the proposed algorithms on the widely used benchmark TRECVID dataset. We will show the experimental results on two proposed kernel machines. (1) the multilabel kernel machine described in Section 2. It exploits the individual concepts and their correlations in a single CML kernel. (2) the multilabel temporal kernel machine described in Section 3. It incorporates the temporal information into CML kernel and models the concept interactions and low-level feature dynamics in CMLT kernel together. We will compare them with other state-of-the-art methods in the first and second paradigms.

4.1 TRECVID Set Description and Experiment Setup

To evaluate the proposed video annotation algorithm, we conduct the experiments on the benchmark TRECVID 2005 data set (http://www-nlpir.nist.gov/projects/trecvid/). This is one of the most widely used data sets by many groups in the area of multimedia concept modeling [Campbell et al. 2006; Chang et al. 2006; Hauptmann et al. 2006]. This data set contains about 170 hours international broadcast news in Arabic, English and Chinese. These news videos are first automatically segmented into 61, 901 subshots [Petersohn 2004]. We select the TRECVID dataset to evaluate our algorithm since it provides us a common platform to compare the performances of different algorithms.

For each subshot, 39 concepts are multilabeled according to LSCOM-Lite annotations [Naphade et al. 2005]. These annotated concepts consist of a wide range of genres, including program category, setting/scene/site, people, object, activity, event, and graphics. Figure 6 illustrates these concepts and

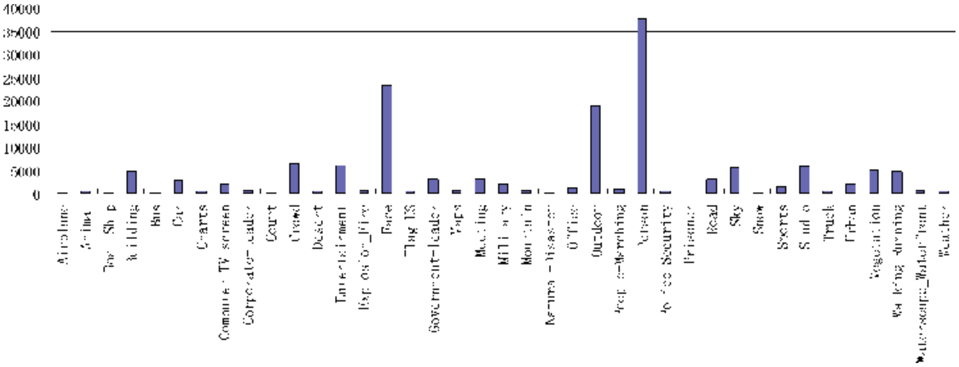

Fig. 6. Video Concepts and their distribution in LSCOM-Lite data set.

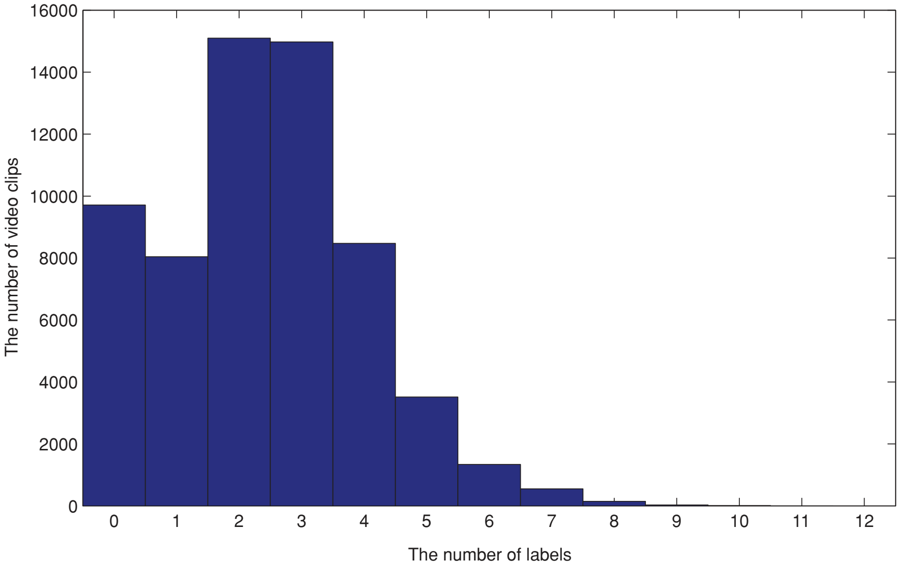

Fig. 7. Distribution of the label numbers in LSCOM-Lite Annotation data set.

the distribution of their numbers in the data set. Intuitively, many of these concepts have significant semantic correlations between each other. Moreover, in the Section 4.2, we also prove that the correlations between different concepts are statistically significant in terms of the normalized mutual information.

Figure 7 illustrates the multilabeling nature of the TRECVID data set. As shown, many subshots (71.32%) have more than one label, and some subshots are even labeled with 11 concepts. Such rich multilabeled subshots in the video data set as well as the significant correlative information between the concepts validate the necessity of exploiting the relationship between the video concepts.

The video data is *sequentially* divided into 3 parts with 65% (40,000 subshots) as training set, 16% (10,000 subshots) as validation set and the remaining 19% (11,901 subshots) as test set. For CBCF, the training set is further split into two parts: one part (32000 subshots) is used for training the individual SVMs in the first detection step, the other part (8000 subshots) is used for training the contextual classifier in the second fusion step. For performance evaluation, we use the official performance metric *Average Precision* (AP) in the TRECVID tasks to evaluate and compare the algorithms on each concept. The AP corresponds to the area under a noninterpolated recall/precision curve and it favors highly ranked relevant subshots. We average the AP over all the 39 concepts to create the mean average precision (MAP), which is the overall evaluation result.

For performance evaluation, we compare our algorithm with two state-of-the-art approaches in first and second paradigms. The first approach, called IndSVM in this section, is the combination of multiple binary encoded SVMs (see the left part of Figure 2), which are trained independently on each concept; the other approach is developed by adding a contextual fusion level on the detection output of the first approach [Godbole and Sarawagi 2004]. In our implementation, we use the SVM for this fusion level. We denote this context-based concept fusion approach as CBCF in this section.

The parameters of the algorithms are determined through a validation process according to their performances on the validation set. For a fair comparison, the results of all the 3 paradigm algorithms reported in this section are the best ones from the chosen parameters. Specifically, two parameters need to be estimated in the proposed CML: the trading-off parameter λ and the Gaussian kernel bandwidth σ of the Gaussian kernel function $\langle \boldsymbol{x}, \tilde{\boldsymbol{x}} \rangle$ in Equations (5) and (24). They are respectively selected from sets {0.5, 1.0, 10, 100} and {0.65, 1.0, 1.5, 2.0} via the validation process. Similarly, the trading-off parameter λ and the Gaussian kernel bandwidth σ in the IndSVM and CBCF are also respectively selected from {0.5, 1.0, 10, 100} and {0.65, 1.0, 1.5, 2.0}, and the best one on the validation set is chosen.

We extract several kinds of low-level features on the keyframes of these subshots [Hua et al. 2006], including

1. Block-wise Color Moment in Lab color space (225D): based on 5-by-5 division of images in Lab color space;
2. Co-occurrence Texture (20D);
3. Wavelet Texture (128D);
4. Edge Distribution Layout (75D);
5. Face (7D): consisting of the face number, face area ratio, the position of the largest face.

Since these features are extracted statically on only keyframes, they are called Static Features (SF), which are different from the Dynamic Features (DF) used in temporal kernel (Equation (23)).

4.2 An Illustration: Interacting Concepts

In Section 2.2, we revealed the connection between the proposed algorithm and GRFs. As has been discussed, the neighborhood set $\mathcal{N}$ is a collection of the interacting concept pairs, and as for CML, this set contains all possible pairs.

However, in practice, some concept pairs may have rather weak interactions, including both positive and negative ones. For example, the concept pairs (airplane, walking running), (people marching, corporate leader) indeed do not have too many correlations, that is to say, the presence/absence of one concept will not contribute to the presence/absence of another concept (i.e., they occur nearly independently). Based on this observation, we only need involve the strongly interacted concept pairs into the set $\mathcal{N}$, and accordingly the kernel function (5) used in CML becomes

$$\langle \theta(\boldsymbol{x}, \boldsymbol{y}), \theta(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}) \rangle = \langle \boldsymbol{x}, \tilde{\boldsymbol{x}} \rangle \sum\nolimits_{1 \le k \le K} \delta[\![y_k = \tilde{y}_k]\!] + \sum\nolimits_{(p,q) \in \mathcal{N}} \delta[\![y_p = \tilde{y}_p]\!] \delta[\![y_q = \tilde{y}_q]\!] \,. \tag{28}$$

The selection of concept pairs can be manually determined by experts or automatically selected by data-driven approaches. In our algorithm, we adopt an automatic selection process in which the expensive

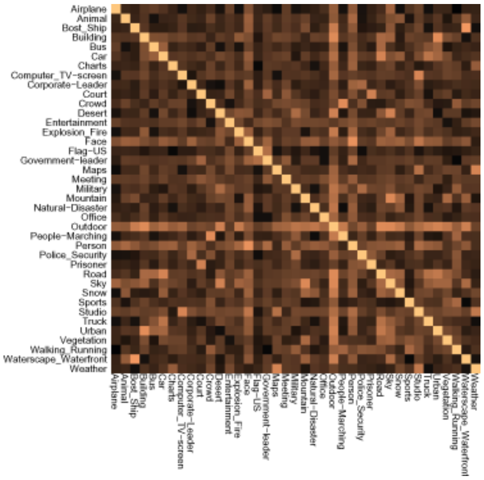

Fig. 8. Each block in this figure illustrates the corresponding normalized mutual information between each pair of the 39 concepts in the LSCOM-Lite annotations data set. These are computed based on the annotations of the development data set in the experiments (see Section 4). The brighter the block is, the large the corresponding mutual information is.

expert labors are not required. First, we use the normalized mutual information [Yao 2003] to measure the correlations of each concept pair (p, q) as

$$NormMI(p,q) = \frac{MI(p,q)}{\min\{H(p), H(q)\}}, \tag{29}$$

where $MI(p,q)$ is the mutual information of the concept p and q, defined by

$$MI(p,q) = \sum\nolimits_{y_p, y_q} P(y_p, y_q) \log \frac{P(y_p, y_q)}{P(y_p)P(y_q)} \tag{30}$$

and $H(p)$ is the marginal entropy of concept p defined by

$$H(p) = -\sum\nolimits_{y_p \in \{+1,-1\}} P(y_p) \log P(y_p). \tag{31}$$

Here the label prior probabilities $P(y_p)$ and $P(y_q)$ can be estimated from the labeled ground-truth of the training dataset. According to the information theory [Yao 2003], the larger the $NormMI(p, q)$ is, the stronger the interaction between concept pair p and q is. Such a normalized measure of concept interrelation has the following advantages:

—It is normalized into the interval [0, 1]: $0 \leq NormMI(p, q) \leq 1$;

—$NormMI(p, q) = 0$ when the concept p and q are statistically independent;

—$NormMI(p, p) = 1$.

These properties are consistent with our intuition on concept correlations, and can be easily proven based on the above definitions. From the above properties, the normalized mutual information is scaled into the interval [0, 1] by the minimum concept entropy. With such a scale, the normalized mutual information only considers the concept correlations, which is irrelevant to the distributions of positive and negative examples of the individual concepts. From the normalized mutual information, the concept pairs whose correlations are larger than a threshold are selected into $\mathcal{N}$. Figure 8 illustrates the normalized mutual information among the 39 concepts in LSCOM-Lite annotation data set. The brighter

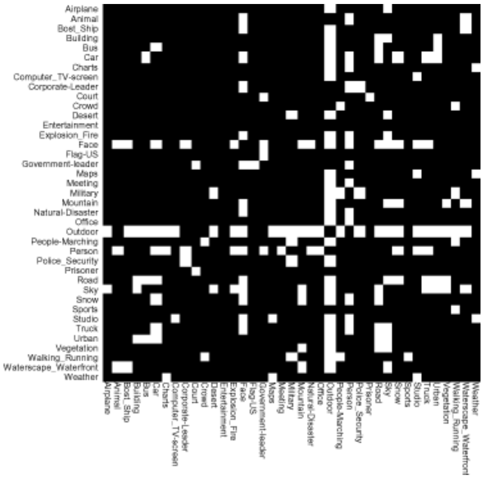

Fig. 9. The selected concept pairs according to the computed normalized mutual information in Figure 8. The white blocks indicate the selected concept pairs with significant correlations.

the grid is, the larger the corresponding normalized mutual information is, and hence the correlation of the concept pair. For example, (“boat ship,” “waterscape waterfront”), (“weather,” “maps”) etc. have larger normalized mutual information. The white dots in Figure 9 represent the selected concept pairs.

4.3 Experiment One: Correlative Multilabel Kernel Machine

In this section, we report experiment results on TRECVID data set. Two different modeling strategies are adopted in the experiments. In the first experiment, all concept pairs are taken into consideration in the model and the kernel function in Equation (5) is adopted. We denote this method by CML(I) in our experiment. In the second one, we adopt the strategy described in Section 4.2, and a subset of the concept pairs is applied based on their interacting significance. Accordingly, the kernel function in Equation (28) is used, and this approach is denoted by CML(II).

4.3.1 *Experiment 1A: Fully Correlative Concepts.* We first conduct experiments of the multilabel method CML (I) with the fully correlative concepts. It considers all possible correlations between the concepts. Figure 10 illustrates the performance of CML(I) compared to that of IndSVM (first paradigm) and CBCF (second paradigm). The following observations can be obtained:

—CML(I) obtains about 15.4% and 12.2% relative improvements on MAP compared to IndSVM and CBCF. Compared to the improvement of CBCF (2%) relative to the baseline IndSVM, such an improvement (i.e., 15.4% relative MAP improvement compared to IndSVM) is significant.

—CML(I) performs the best on 28 of the all 39 concepts. Some of the improvements are significant, such as “office” (477% better than InidSVM and 260% better than CBCF), “people-marching” (68% better than IndSVM and 160% better than CBCF), “walking running” (55% better than IndSVM and 48% better than CBCF).

—CML(I) deteriorates on some concepts compared to IndSVM and CBCF. For example, it has 12% and 14% deterioration on “snow” respectively and 11% and 17% deterioration on “bus” respectively. As discussed in Section 4.2, the performance deterioration is due to insignificant concept relations. Next subsection will present CML(II), which solves this deterioration problem and obtains a more consistent and robust performance improvement.

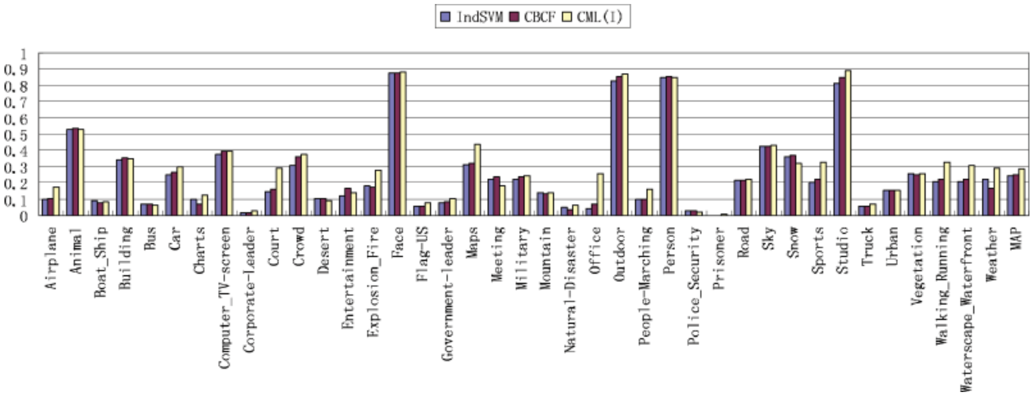

Fig. 10. The performance comparison of IndSVM, CBCF and CML(I).

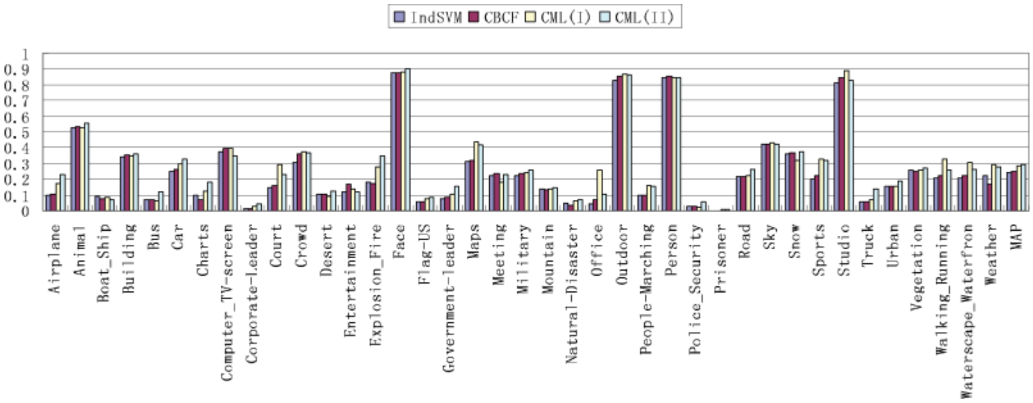

Fig. 11. The performance comparison of IndSVM, CBCF, CML (I) and CML(II).

4.3.2 *Experiment 1B: Partially Correlative Concepts.* Following the proposed approach in Section 4.2, the deterioration problem can be solved by removing concept pairs with insignificant correlations.

Figure 8 illustrates the normalized mutual entropy among all concepts. They are computed on the development set which includes training set and validation set, but does NOT include the test set. The average normalized mutual information entropy is $Avg_{EN} = 0.02$. An important aspect of a good algorithm is if its parameters can be determined automatically. Following such a principle, the threshold Th_{EN} is automatically determined to be $Th_{EN} = 2Avg_{EN}$ such that any concept pairs whose normalized mutual entropy less than Th_{EN} are removed. Figure 9 shows these selected concept pairs. As we can see, these preserved concept pairs either have intuitive semantic correlations for example, "waterscape waterfront" and "boat ship" or statistically tend to co-occur in the news broadcast videos, for example, "maps" and "weather" in weather forecast video subshots.

Figure 11 illustrates the performance of CML(II) with these selected concept pairs compared to IndSVM, CBCF and CML(I). We can find

—CML(II) has the best overall performance compared to the other algorithms. It outperforms IndSVM, CBCF and CML(I) by 17%, 14% and 2%, respectively.

—Furthermore, CML(II) has a more consistent and robust performance improvement over all 39 concepts compared to IndSVM and CBCF. For example, on "bus" and "snow," CML(I) gave worse performance than IndSVM and CBCF. In the contrary, CML(II) gains about 71% and 3% improvement compared to IndSVM and 58% and 1% improvement compared to CBCF with no deterioration.

In summary, CML(II) is the best approach in terms of its best overall MAP improvement as well as its consistent and robust performance on the diverse 39 concepts. As we can find in Figure 11, CML(II) has improved the performance on 24 of all the 39 concepts compared to IndSVM, CBCF, CML(I). Most of these improvements attribute to the compensation of complementary information between different labels, for example, the annotation of "Bus" can obtain the complementary information from the annotation of "road." However, there also exist fewer concepts whose annotation performance are reduced. For example, CML(II) has a degradation on "Office" compared to CML(I). It is probably caused by removal of too many related concept pairs relevant to "Office." According to Figure 11, we only retain two concept pairs "Office/Outdoor" and "Office/Person" while the related pairs, such as "Office/Face," "Office/Meeting," and "Office/Corporate-Leader," have been removed. These removed pairs potentially contribute a lot to the annotation of "Office." It indicates the trade-off between removing useless concept interactions and preserving significant interactions is still a difficult problem. A promising solution to this problem is to involve human experts' prior knowledge to determine which concept interactions are mostly helpful to concept annotation. We leave the study of such a human-centered method in the future work.

4.4 Experiment Two: Correlative Multilabel Temporal Kernel Machine

In this section, we evaluate the proposed CMLT kernel method in Section 3. As aforementioned, this method can further capture the temporal information of video sequences. Compared to the formal CML method that only extracts static features on the keyframes of video subshots, this CMLT kernel machine can capture the dynamic features contained in the temporal patterns of the videos. Such dynamic patterns are important sources for improving the discrimination between different video concepts.

As depicted in Section 3, all subshots are regarded as sequences of video frames, and the low-level features are extracted on these frame sequences rather than only keyframes of each subshot. To accelerate the feature extraction and model learning, we do not extract features on every frame. Instead, we only extract the features at the rate of one frame per second. These extracted features are then used to train the HMM for each subshot. In more detail, a universal reference model is first trained on 5000 video subshots which are randomly selected from the training set. Then for each subshot, a HMM is adapted from this URM according to Equation (27) and EM algorithm (see Section 3.2 for detail). The low-level features extracted on the video frames are the same as the static features used in experiment one. However, since they are extracted on frame sequences to train a dynamic model, we call them *Dynamic Features* (DF) (see Figure 12). It is worthy of noting that URM only provides a background model and it is not used to obtain temporal kernel directly. Actually, the temporal information in each subshot is encoded into the adapted HMMs through MAP adaption instead. As stated in Section 3.2, the effect of URM is to make the underlying states in each adapted HMM have a reasonable correspondence between them, so that the obtained KLD upper bound between two HMMs is tight enough.

For the sake of the fair comparison, we follow the same experiment settings in experiment one. Table I illustrates the performance of CMLT kernel method with the comparisons of IndSVM, CBCF, CML(I), CML(II). From these results, we can find

—The CMLT machine has the best overall performance in terms of MAP. It outperforms the IndSVM, CBCF, CML(I) and CML(II) by 35.0%, 31.3%, 17.0%, 14.7%.

—CMLT gains the best performance on 30 concepts out of the whole 39 concepts.

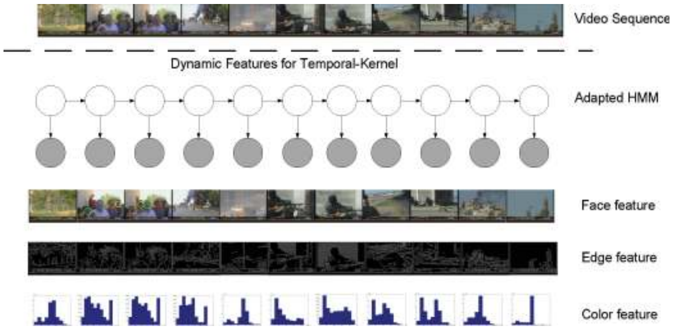

Fig. 12. The dynamic features used in temporal kernel: the low-level features are extracted at the rate of one frame per second, and these extracted features are then used to train an adapted HMM from URM. It is contrast to the static features that are extracted only on keyframes of the subshots.

Take a more insight observation into the CMLT result, we can find on four event-related concepts, that is, "Explosion_Fire," "Natural-Disaster," "People-Marching," "Walking_Running," the CMLT significantly outperforms the other four methods. This takes advantage of the temporal dynamics contained in these event concepts. On the other hand, we can also observe performance degradation on few concepts such as "Computer_TV-screen" and "Entertainment." It is due to the relatively weak temporal information in these video sequences.

Finally, we would like to have a brief discussion about the computational cost of the proposed CML methods. In fact, under the different parameter settings, the computational cost is different largely. But in general, as for IndSVM and CBCF, the models of each concept are independent without coupled relation with each other, so they can be trained in parallel, that is, the models of different concepts can be trained at the same time. Therefore, the computing time needed is much less than CML in which the modeling of the whole concept set is conducted in a coupled manner and thus is unable to be operated in parallel. In our experiment, the speed of CML is about 25 times slower than IndSVM and CBCF. Therefore how to accelerate the computation speed of CML will be the focus of our future work. Specifically, an online algorithm can be a good choice to solve the efficiency issue on the large-scale problem; that is, the online algorithm can incrementally learn the multilabeled prediction model so that the model parameters can be updated once one or a batch training samples arrive, instead of learning these parameters after accumulating all training samples. We believe such an online algorithm can greatly accelerate the training process in real time.

5. CONCLUSION AND FUTURE WORK

We propose a *Correlative Multilabel* (CML) kernel machine in this paper to leverage the label correlations to help infer the video concepts. It exploits the individual concepts and their correlations in a single formulation. We analogize the CML to the *Gibbs Random Field* (GRF) to justify that CML can simultaneously model the individual concepts and their correlations from a statistical perspective. Based on this GRF formulation, we give an efficient inference algorithm to predict the labels given an input video sequence. The experimental results show the CML and improve the overall annotation performance compared to the state-of-the-art algorithms in the other two paradigms for video annotation. Moreover, they can consistently improve the annotation accuracy over the most of concepts with slight degradation on very few concepts. As we have pointed out, this degradation can be avoided by more sophisticated interactive method to incorporate experts' knowledge of concept ontology structure to preserve the most significant concept interactions. This will be the topic of our future work. Another

Table I.
The average precision over 39 LSCOM-lite concepts for the five algorithms: IndSVM, CBCF, CML(I), CML(II), CMLT. The CMLT gains the best over performance of these algorithm, and it also outperforms the other four algorithms on 30 out of 39 concepts. The bold indicates the best approach for each concept in this table.

	IndSVM	CBCF	CML(I)	CML(II)	CMLT
Airplane	0.1005	0.1019	0.1712	0.2325	**0.2563**
Animal	0.5265	0.5336	0.5302	0.55824	**0.7193**
Boat_Ship	**0.087**	0.0798	0.0849	0.0707	0.0779
Building	0.3375	0.3538	0.3486	0.3585	**0.3952**
Bus	0.0669	0.0724	0.0602	0.1147	**0.1706**
Car	0.2469	0.2673	0.2983	0.3296	**0.4185**
Charts	0.0981	0.0709	0.1277	0.182	**0.2558**
Computer_TV-screen	0.3773	0.3927	**0.3976**	0.3438	0.379
Corporate-Leader	0.0112	0.014	0.03	0.0438	**0.0483**
Court	0.1462	0.1568	**0.294**	0.232	0.2558
Crowd	0.3073	0.3598	0.3775	0.3676	**0.4053**
Desert	0.1047	0.1053	0.0902	0.125	**0.1378**
Entertainment	0.1174	**0.1687**	0.14	0.1171	0.1291
Explosion_Fire	0.1773	0.1768	0.2755	0.3447	**0.38**
Face	0.8779	0.8782	0.8854	0.9062	**0.9762**
Flag-US	0.0571	0.0563	0.0759	0.084	**0.0926**
Government-leader	0.0774	0.0838	0.1029	0.1515	**0.167**
Maps	0.3147	0.3206	0.4347	0.4156	**0.5228**
Meeting	0.2208	0.2391	0.183	0.232	**0.2558**
Military	0.2202	0.2337	0.2405	**0.2571**	0.2394
Mountain	0.1367	0.135	0.1397	0.148	**0.1632**
Natural-Disaster	0.0462	0.0381	0.0633	0.0664	**0.0932**
Office	0.044	0.0706	**0.2541**	0.1053	0.1161
Outdoor	0.823	0.8517	**0.8695**	0.8607	0.8166
People-Marching	0.095	0.0998	0.1595	0.1561	**0.2949**
Person	0.8441	0.8535	0.8453	0.844	**0.9856**
Police_Security	0.0301	0.0253	0.02	0.058	**0.0639**
Prisoner	0.0026	0.0016	0.0039	0.0096	**0.0106**
Road	0.2169	0.2158	0.2249	0.2656	**0.2928**
Sky	0.4204	0.4261	0.4281	0.4213	**0.4645**
Snow	0.3625	0.37	0.3179	0.374	**0.4123**
Sports	0.2025	0.2194	0.329	0.3226	**0.3998**
Studio	0.8109	0.8448	0.889	0.8283	**0.9132**
Truck	0.0552	0.0529	0.0727	0.1381	**0.1523**
Urban	0.151	0.1528	0.1517	0.1861	**0.2052**
Vegetation	0.2596	0.2511	0.2537	0.2675	**0.2949**
Walking_Running	0.2094	0.2188	0.3251	0.2565	**0.3828**
Waterscape_Waterfront	0.2049	0.2219	0.3055	0.2642	**0.2913**
Weather	0.2200	0.1690	0.2898	0.2765	**0.3364**
Mean average precision	0.2463	0.2534	0.2843	0.2901	**0.3326**

promising extension to CML and CMLT is to design an online learning algorithm to learn the multilabeled model. It can greatly accelerate the learning process so that once one or a batch of new training samples arrive, the model parameters can be updated in time with no need of training the model after all the samples are accumulated. We believe such an online extension can make the algorithm applicable in many real-time environments such as Web application.

Besides the correlative multilabel model, we also introduce a temporal kernel into the CML formulation to form a *Correlative Multilabel Temporal* (CMLT) kernel machine. This new kernel method takes into account not only feature dynamics but also concept interactions in an integrated manner. It obeys the principle of least commitment without any extra step that induces propagate errors to its consecutive step so that a better performance can be expected. Experiments also show this temporal information used in CMLT can serve as an important resource to improve the annotation accuracy on many event-related concepts, such as "People-Marching," and "Explosion_Fire." The event detection in video sequences have attracted many attentions, and we believe this temporal kernel can contribute to revealing if and how temporal information be utilized to detect various event concepts of videos.

APPENDIX A—LEARNING THE CLASSIFICATION FUNCTION

In this section, we will introduce how to train the classification model (1) with the presented kernel (5). The procedure follows a similar derivation as in the conventional SVM (details about SVM can be found in Cristianini and Shawe-Taylor [2000]) and in particular one of its variants for the structural output spaces [Tsochantaridis et al. 2004]. Given an example $\boldsymbol{x}_i$ and its label vector $\boldsymbol{y}_i$ from the training set $\{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^n$, according to Equations (1) and (2), a misclassification occurs when we have

$$\Delta F_i(\boldsymbol{y}) \triangleq F(\boldsymbol{x}_i, \boldsymbol{y}_i) - F(\boldsymbol{x}_i, \boldsymbol{y}) = \langle \boldsymbol{w}, \Delta\theta_i(\boldsymbol{y})\rangle \leq 0, \forall \boldsymbol{y} \neq \boldsymbol{y}_i, \boldsymbol{y} \in \mathcal{Y}, \tag{32}$$

where $\Delta\theta_i(\boldsymbol{y}) = \theta(\boldsymbol{x}_i, \boldsymbol{y}_i) - \theta(\boldsymbol{x}_i, \boldsymbol{y})$. Therefore, the empirical prediction risk on training set wrt the parameter $\boldsymbol{w}$ can be expressed as

$$\hat{R}(\{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^n; \boldsymbol{w}) = \frac{1}{n}\sum\nolimits_{i=1}^n \sum\nolimits_{\boldsymbol{y}\neq\boldsymbol{y}_i, \boldsymbol{y}\in\mathcal{Y}} \ell(\boldsymbol{x}_i, \boldsymbol{y}; \boldsymbol{w}), \tag{33}$$

where $\ell(\boldsymbol{x}_i, \boldsymbol{y}; \boldsymbol{w})$ is a loss function counting the errors as

$$\ell(\boldsymbol{x}_i, \boldsymbol{y}; \boldsymbol{w}) = \begin{cases} 1 \text{ if } \langle \boldsymbol{w}, \Delta\theta_i(\boldsymbol{y})\rangle \leq 0, \forall \boldsymbol{y} \neq \boldsymbol{y}_i, \boldsymbol{y} \in \mathcal{Y}; \\ 0 \text{ if } \langle \boldsymbol{w}, \Delta\theta_i(\boldsymbol{y})\rangle > 0, \forall \boldsymbol{y} \neq \boldsymbol{y}_i, \boldsymbol{y} \in \mathcal{Y}. \end{cases} \tag{34}$$

Our goal is to find a parameter $\boldsymbol{w}$ that minimizes the empirical error $\hat{R}(\{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^n; \boldsymbol{w})$. Considering the computational efficiency, in practice, we use the following convex loss which upper bounds $\ell(\boldsymbol{x}_i, \boldsymbol{y}; \boldsymbol{w})$ to avoid directly minimize the step-function loss:

$$\ell_h(\boldsymbol{x}_i, \boldsymbol{y}; \boldsymbol{w}) = (1 - \langle \boldsymbol{w}, \Delta\theta_i(\boldsymbol{y})\rangle)_+, \tag{35}$$

where $(\cdot)_+$ is a hinge loss in classification. Correspondingly, we can now define the following empirical hinge risk which upper bounds $\hat{R}$ $(\{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^n; \boldsymbol{w})$:

$$\hat{R}_h(\{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^n; \boldsymbol{w}) = \frac{1}{n}\sum\nolimits_{i=1}^n \sum\nolimits_{\boldsymbol{y}\neq\boldsymbol{y}_i, \boldsymbol{y}\in\mathcal{Y}} \ell_h(\boldsymbol{x}_i, \boldsymbol{y}; \boldsymbol{w}). \tag{36}$$

Accordingly, we can formulate a regularized version of $\hat{R}_h(\{\boldsymbol{x}_i , \boldsymbol{y}_i\}_{i=1}^n; \boldsymbol{w})$ that minimizes an appropriate combination of the empirical error and a regularization term $\Omega(||\boldsymbol{w}||^2)$ to avoid overfitting of the learned model. That is,

$$\min_{\boldsymbol{w}} \left\{\hat{R}_h(\{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^n; \boldsymbol{w}) + \lambda \cdot \Omega(||\boldsymbol{w}||^2)\right\}, \tag{37}$$

where Ω is a strictly monotonically increasing function, and λ is a parameter trading off between the empirical risk and the regularizer. As indicated in Cristianini and Shawe-Taylor [2000], such a regularization term can give some smoothness to the obtained function so that the nearby mapped $\theta(\boldsymbol{x}, \boldsymbol{y})$, $\theta(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}})$ have the similar function value $F(\theta(\boldsymbol{x}, \boldsymbol{y}); \mathbf{w})$, $F(\theta(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}); \mathbf{w})$. Such a local smoothness assumption is intuitive and can relieve the negative influence of the noise training data.

In practice, the above optimization problem can be solved by reducing it to a convex quadratic problem. Similar to what is done in SVMs [Cristianini and Shawe-Taylor 2000], by introducing a slack variable $\xi_i(\boldsymbol{y})$ for each pair $(\boldsymbol{x}_i, \boldsymbol{y})$, the optimization formulation in (37) can be rewritten as

$$\begin{aligned}&\min_{\boldsymbol{w}} \frac{1}{2}||\boldsymbol{w}||^2 + \frac{\lambda}{n} \cdot \sum\nolimits_{i=1}^{n} \sum\nolimits_{\boldsymbol{y}\neq\boldsymbol{y}_i, \boldsymbol{y}\in\mathcal{Y}} \xi_i(\boldsymbol{y}) \\ &s.t. \langle \boldsymbol{w}, \Delta\theta_i(\boldsymbol{y})\rangle \geq 1 - \xi_i(\boldsymbol{y}), \xi_i(\boldsymbol{y}) \geq 0 \quad \boldsymbol{y} \neq \boldsymbol{y}_i, \boldsymbol{y} \in \mathcal{Y}\end{aligned} \tag{38}$$

On introducing Lagrange multipliers $\alpha_i(\boldsymbol{y})$ into the above inequalities and formulating the Lagrangian dual according to Karush-Kuhn-Tucker (KKT) theorem [Boyd and Vandenberghe 2004], the above problem further reduces to the following convex quadratic problem (QP):

$$\begin{aligned}&\max_{\alpha} \sum_{i,\boldsymbol{y}\neq\boldsymbol{y}_i} \alpha_i(\boldsymbol{y}) - \frac{1}{2} \sum_{i,\boldsymbol{y}\neq\boldsymbol{y}_i} \sum_{j,\tilde{\boldsymbol{y}}\neq\boldsymbol{y}_j} \alpha_i(\boldsymbol{y})\alpha_j(\tilde{\boldsymbol{y}})\langle \Delta\theta_i(\boldsymbol{y}), \Delta\theta_j(\tilde{\boldsymbol{y}})\rangle \\ &s.t.\, 0 \leq \sum\nolimits_{\boldsymbol{y}\neq\boldsymbol{y}_i, \boldsymbol{y}\in\mathcal{Y}} \alpha_i(\boldsymbol{y}) \leq \frac{\lambda}{n}, \boldsymbol{y} \neq \boldsymbol{y}_i, \boldsymbol{y} \in \mathcal{Y}, 1 \leq i \leq n\end{aligned} \tag{39}$$

and the equality

$$\boldsymbol{w} = \sum\nolimits_{1\leq i\leq n, \boldsymbol{y}\in\mathcal{Y}} \alpha_i(\boldsymbol{y})\Delta\theta_i(\boldsymbol{y}) \tag{40}$$

Different from those dual variables in the conventional SVMs which only depend on the training data of observation and the associated label pairs $(\boldsymbol{x}_i, \boldsymbol{y}_i)$, $1 \leq i \leq n$, the Lagrangian duals in (39) depend on all assignment of labels $\boldsymbol{y}$, which are not limited to the true label of $\boldsymbol{y}_i$. We can iteratively find the active constraints and the associated label variable $\boldsymbol{y}^*$ which most violates the constraints in (35) as $\boldsymbol{y}^* = \arg\max_{\boldsymbol{y}\neq\boldsymbol{y}_i} F(\boldsymbol{x}_i, \boldsymbol{y}; \boldsymbol{w})$ and $\Delta F_i(\boldsymbol{y}^*) < 1$. An active set is maintained for these corresponding active dual variables $\alpha_i(\boldsymbol{y}^*)$, and $\boldsymbol{w}$ is optimized over this set during each iteration using commonly available QP solvers (e.g., SMO [Cristianini and Shawe-Taylor 2000]).

APPENDIX B—PROOF OF KULLBACK-LEIBLER UPPER BOUNDS

Here we prove the upper bounds (20), (21) of KLD between two HMMs Θ, $\tilde{\Theta}$. This proof is given by Do [2003]. The approximation is motivated from the following upper bound that is based on the chain rule for relative entropy [Cover and Thomas 1991]:

LEMMA 1. *Given two mixture distributions $f = \sum_{i=1}^{L} w_i f_i$ and $g = \sum_{i=1}^{L} v_i g_i$, the KLD between them is upper bounded by*

$$D_{KL}(f||g) \leq D_{KL}(w||v) + \sum\nolimits_{i=1}^{L} w_i D_{KL}(f_i||g_i), \tag{41}$$

where $D_{KL}(w||v) = \sum_{i=1}^{L} w_i \log \frac{w_i}{v_i}$. This inequality directly follows the log-sum inequality (see pp. 31 of Cover and Thomas [1991]).

Let backward variables $\beta_t(i) = P(o_t o_{t+1} \cdots o_{\mathcal{T}} | s_t = i, \Theta)$ denote the probability that the sequence $o_t o_{t+1} \cdots o_{\mathcal{T}}$ is observed given the current state s_t is i and $\pi = [\,\pi_1\ \pi_2\ \cdots\ \pi_Q\,]^T$ denote the initial state distribution. Thus the distribution of the whole observation sequences can be computed by the Baum-Welch algorithm [Rabiner 1989] as

$$P(O|\Theta) = \sum_{i=1}^{Q} \pi_i \beta_t(i). \tag{42}$$

Therefore based on Lemma 1, the KLD between two HMMs $\Theta, \tilde{\Theta}$ can be computed from Lemma 1 as

$$\begin{aligned} D_{KL}(\Theta||\tilde{\Theta}) &= D_{KL}\left(\sum\nolimits_{i=1}^{Q} \pi_i \cdot \beta_t(i) || \sum\nolimits_{i=1}^{Q} \tilde{\pi}_i \cdot \tilde{\beta}_t(i)\right) \\ &\leq D_{KL}(\pi||\tilde{\pi}) + \sum\nolimits_{i=1}^{Q} \pi_i \cdot D_{KL}\left(\beta_t(i)||\tilde{\beta}_t(i)\right). \end{aligned} \tag{43}$$

The term $D_{KL}(\beta_t(i)||\tilde{\beta}_t(i))$ can be computed by utilizing the following recursive formulation:

$$\beta_t(i) = b_i(o_t) \sum_{j=1}^{Q} a_{i,j} \beta_{t+1}(j). \tag{44}$$

Thus

$$D_{KL}\left(\beta_t(i)||\tilde{\beta}_t(i)\right) \leq D_{KL}(b_i||\tilde{b_i}) + D_{KL}(a_{i,\cdot}||\tilde{a}_{i,\cdot}) + \sum_{i=1}^{Q} a_{i,j} D_{KL}\left(\beta_{t+1}(j)||\tilde{\beta}_{t+1}(j)\right). \tag{45}$$

We can define $D_t = [D_t^1 D_t^2 \cdots D_t^Q]^T$ with $D_t^i = D_{KL}(\beta_t(i)||\tilde{\beta}_t(i))$ and $C = [C_1 C_2 \cdots C_Q]^T$ with $C_i = D_{KL}(b_i||\tilde{b_i}) + D_{KL}(a_{i,\cdot}||\tilde{a}_{i,\cdot})$. Thus Equations (43) and (45) can be rewritten as

$$\begin{aligned} &D_{KL}(\Theta||\tilde{\Theta}) \leq D_{KL}(\pi||\tilde{\pi}) + \pi^T D_1 \\ &D_t \leq C + A \cdot D_{t+1}, \end{aligned} \tag{46}$$

where $A = (a_{i,j})_{Q\times Q}$ is the transition matrix. Therefore, we have

$$D_{KL}(\Theta||\tilde{\Theta}) \leq D_{KL}(\pi||\tilde{\pi}) + \pi^T \left(\sum_{t=1}^{\mathcal{T}-1} A^{t-1} C + A^{\mathcal{T}-1} D\right). \tag{47}$$

Assume that the model Θ is stationary so a stationary distribution γ exists, that is,

$$\begin{aligned} \gamma^T A &= \gamma^T \\ \lim_{t\to\infty} \pi^T A^t &= \gamma^T. \end{aligned} \tag{48}$$

Therefore, combining Equations (47) and (48), the KLD rate between two HMMs can be

$$\begin{aligned} \widehat{D}_{KL}(\Theta||\tilde{\Theta}) &= \lim_{\mathcal{T}\to\infty} \frac{1}{\mathcal{T}} D_{KL}(\Theta||\tilde{\Theta}) \\ &\leq \gamma^T C = \sum_{i=1}^{Q} \gamma_i \{D_{KL}(b_i||\tilde{b_i}) + D_{KL}(a_{i,\cdot}||\tilde{a}_{i,\cdot})\}. \end{aligned} \tag{49}$$

Similarly, we can obtain the reverse KLD rate as

$$\widehat{D}_{KL}(\tilde{\Theta}||\Theta) \leq \sum_{i=1}^{Q} \tilde{\gamma}_i \{D_{KL}(\tilde{b}||b_i) + D_{KL}(\tilde{a}_{i,\cdot}||a_{i,\cdot})\}, \tag{50}$$

where $\tilde{\gamma}$ is the stationary distribution of the model $\tilde{\Theta}$.

REFERENCES

BERG, B. A. 2004. *Markov Chain Monte Carlo Simulations and Their Statistical Analysis*. World Scientific.

BOYD, S., VANDENBERGHE, L. 2004. *Convex Optimization*. Cambridge University Press.

CAMPBELL, M., ET AL. 2006. Ibm research trecvid-2006 video retrieval system. *TREC Video Retrieval Evaluation (TRECVID) Proceedings*.

Chang, S.-F., et al. 2006. Columbia university trecvid-2006 video search and high-level feature extraction. In *TREC Video Retrieval Evaluation (TRECVID) Proceedings*.

Cover, T. and Thomas, J. 1991. *Elements of Information Theory*. John Wiley and Sons, New York, NY.

Cristianini, N. and Shawe-Taylor, J. 2000. *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge University Press.

Do, M. 2003. Fast approximation of kullback-leibler distance for dependence trees and hidden markov models. *IEEE Signal Process. Lett. 10*, 4, 115–118.

Ebadollahi, S., Xie, L., Chang, S.-F., and Smith, J. R. 2006. Visual event detection using multidimensional concept dynamics. In *Proceedings of the IEEE International Conference on Multimedia and Expo*.

Gauvain, J.-L. and Lee, C.-H. 1994. Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. *IEEE Trans. Speech Audio Process. 2*, 2, 291–298.

Godbole, S. and Sarawagi, S. 2004. Discriminative methods for multi-labeled classification. In *Proceedings of the Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*.

Goldberger, J. and Aronowitz, H. 2005. A distance measure between gmms based on the unscented transform and its application to speaker recognition. In *Proceedings of the International Conference on Spoken Language Processes*.

Hauptmann, A. G., Chen, M.-Y., and Christel, M. 2004. Confounded expectations: Informedia at TRECVID 2004. In *TREC Video Retrieval Evaluation (TRECVID) Proceedings*.

Hauptmann, A. G., et al. 2006. Multi-lingual broadcast news retrieval. In *TREC Video Retrieval Evaluation (TRECVID) Procedings*.

Hauptmann, A. G., Yan, R., Lin, W.-H., Christel, M., and Wactlar, H. 2007. Can high-level concepts fill the semantic gap in video retrieval? A case study with broadcast news. *IEEE Trans. Multimed. 9*, 5, 958–966.

Hua, X.-S., Mei, T., Lai, W., Wang, M., Tang, J., Qi, G.-J., Li, L., and Gu, Z. 2006. Microsoft reseach asia trecvid 2006 high-level feature extraction and rushes exploitation. In *TREC Video Retrieval Evaluation (TRECVID) Proceedings*.

Jiang, W., Chang, S.-F., and Loui, A. 2006. Active concept-based concept fusion with partial user labels. In *Proceedings of the IEEE International Conference on Image Processing*.

Jiang, Y.-G., Ngo, C.-W., and Yang, J. 2007. Towards optimal bag-of-features for object categorization and semantic video retrieval. In *Proceedings of the ACM International Conference on Image and Video Retrieval*.

Koskela, M., Smeaton, A., and Laaksonen, J. 2007. Measuring concept similarities in multimedia ontologies: analysis and evaluations. *IEEE Trans. Multimed. 9*, 5, 912–922.

Kumar, S. and Hebert, M. 2003. Discriminative random fields: A discriminative framework for contextual interaction in classification. In *Proceedings of the IEEE International Conference on Machine Learning*.

Lafferty, J., McCallum, A., and Pereira, F. 2001. Conditional random fields: Probabilistic models for segmentation and labeling sequence data. In *Proceedings of the International Conference on Machine Learning*.

Liu, P., Soong, F. K., and Zhou, J.-L. 2007. Divergence-based similarity measure for spoken document retrieval. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*.

Marr, D. 1982. *Vision*. W. H. Freeman and Company.

Naphade, M. R., Kozintsev, I., and Huang, T. 2002. Factor graph framework for semantic video indexing. *IEEE Trans. CSVT 12*, 1 (Jan.).

Naphade, M. R., Smith, J., Tesic, J., Chang, S.-F., Hsu, W., Kennedy, L., Hauptmann, A. G., and Curtis, J. 2006. Large-scale concept ontology for multimedia. *IEEE Trans. Multimed. 13*, 3, 86–91.

Naphade, M. R. 2002. Statistical techniques in video data management. In *Proceedings of the IEEE Workshop on Multimedia Signal Processing*.

Naphade, M. R., Kennedy, L., Kender, J. R., Chang, S.-F., Smith, J. R., Over, P., and Hauptmann, A. G. 2005. A light scale concept ontology for multimedia understanding for TRECVID 2005. IBM Research Report RC23612 (W0505-104).

Nigam, K., Lafferty, J., and McCallum, A. 1999. Using maximum entropy for text classification. In *Proceedings of the IJCAI-99 Workshop on Machine Learning for Information Filtering*. 61–67.

Petersohn, C. 2004. Fraunhofer hhi at trecvid 2004: shot boundary detection system. In *TREC Video Retrieval Evaluation (TRECVID) Proceedings*.

Rabiner, L. R. 1989. A tutorial on hidden markov models and selected applications in speech recognition. *Proc. IEEE 77*, 2, 257–286.

Smeaton, A., Over, P., and Kraaij, W. 2006. Evaluation campaigns and trecvid. In *Proceedings of the ACM Multimedia Information Retrieval Conference*. 321–330.

SMITH, J. R. AND NAPHADE, M. R. 2003. Multimedia semantic indexing using model vectors. In *Proceedings of the IEEE Internaional Conference on Multimedia and Expo*.

SNOEK, C., WORRING, M., GEUSEBROEK, J., KOELMA, D., SEINSTRA, F., AND SMEULDERS, A. 2006. The semantic pathfinder: Using an authoring metaphor for generic multimedia indexing. *IEEE Trans. Patt. Anal. Mach. Intell. 28*, 10, 1678–1689.

SNOEK, C. G. M., WORRING, M., GEMERT, J. C., GEUSEBROEK, J.-M., AND SMEULDERS, A. W. M. 2006. The challenge problem for automated detection of 101 semantic concepts in multimedia. In *Proceedings of the ACM Internaional Conference on Multimedia*. 421–430.

TANG, J., HUA, X.-S., QI, G.-J., WANG, M., MEI, T., AND WU, X. 2007. Structure-sensitive manifold ranking for video concept detection. In *Proceedings of the ACM Internaional Conference on Multimedia*.

TSOCHANTARIDIS, I., HOFMANN, T., JOACHIMS, T., AND ALTUN, Y. 2004. Support vector machine learning for intedependent and structured output spaces. In *Proceedings of the Internaional Conference on Machine Learning*.

WANG, D., LIU, X., LUO, L., LI, J., AND ZHANG, B. 2007. Video diver: Generic video indexing with diverse features. In *Proceedings of the ACM Conference on Multimedia Information Retrieval*.

WANG, T., LI, J., DIAO, Q., HU, W., ZHANG, Y., AND DULONG, C. 2006. Semantic event detection using conditional random fields. In *Proceedings of the IEEE Computer Vision and Pattern Recognition Workshop*.

WINKLER, G. 1995. *Image Analysis, Random Fields and Dynamic Monte Carlo Methods: A Mathematical Introduction*. Springer-Verlag, Berlin, Heidelberg.

WU, Y., TSENG, B. L., AND SMITH, J. R. 2004. Ontology-based multi-classification learning for video concept detection. In *Proceedings of the IEEE Internaional Conference on Multimedia and Expo*.

XIE, L. AND CHANG, S.-F. 2002. Structural analysis of soccer video with hidden markov models. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*.

YAN, R., CHEN, M.-Y., AND HAUPTMANN, A. G. 2006. Discriminative random fields: A discriminative framework for contextual interaction in classification. In *Proceedings of the IEEE Internaional Conference on Multimedia and Expo*.

YANAGAWA, A., CHANG, S.-F., KENNEDY, L., AND HSU, W. 2007. Columbia university's baseline detectors for 374 lscom semantic visual concepts. Tech. Rep. 222-2006-8, Columbia University ADVENT Technical Report. March. 20.

YAO, Y. Y. 2003. *Entropy Measures, Maximum Entropy Principle, and Emerging Applications*. Springer, Chapter Information-theoretic measures for knowledge discovery and data mining, 115–136.

ZHA, Z.-J., MEI, T., HUA, X.-S., QI, G.-J., AND WANG, Z. 2007. Refining video annotation by exploiting pairwise concurrent relation. In *Proceedings of the ACM International Conference on Multimedia*.

ZHANG, H., BERG, A. C., MAIRE, M., AND MALIK, J. 2006. Svm-knn: discriminative nearest neighbor classification for visual category recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Received January 2008; revised May 2008; accepted July 2008