Efficient Genre-Specific Semantic Video Indexing

Jun Wu and Marcel Worring, *Member, IEEE*

***Abstract*—Large video collections such as YouTube contain many different video genres, while in many applications the user might be interested in one or two specific video genres only. Thus, when users are querying the system with a specific semantic concept like *AnchorPerson*, and *MovieStars*, they are likely aiming a genre specific instantiation of this concept. Existing methods treat this problem as a classical learning problem leading to unnecessarily complex models. We propose a framework to detect visual-based genre-specific concepts in a more efficient and accurate way. We do so by using a two-step framework distinguishing two different levels. Genre-specific concept models are trained based on a training set with data labeled at video level for genres and at shot level for semantic concepts. In the classification stage, video genre classification is applied first to reduce the entire data set to a relatively small subset. Then, the genre-specific concept models are applied to this subset only. Experiments have been conducted on a small 28-h data set for genre-specific concept detection and a 4168-h (80 031 videos) benchmark data set for genre-specific topic search. Experimental results show that our proposed two-step method is more efficient and effective than existing methods which do not consider the different semantic levels between video genres and semantic concepts for both the indexing and the search tasks. When filtering out 80% of the data set, the average performance loss is about 11.3% for genre-specific concept detection and 31.5% for genre-specific topic search, while the processing speed increases hundreds of times for different video genres.**

***Index Terms*—Efficiency, genre classification, genre-specific concept detection, semantic indexing.**

I. Introduction

SEARCHING video content is difficult, because the volume of video increases rapidly while we lack the proper tools to handle large-scale video sets. Search in popular search engines is still by tags only. However, tags are subjective and noisy, and, in many cases, they are not reflecting the content. Semantic indexes derived from video content have been proven to be a convenient way of accessing video [1], but semantic concepts can vary widely in visual appearance, and efficient processing is still a difficult task. Though in some cases the audio or metadata can provide additional distinguishing information, they are often not available, and their utility is still limited. Therefore, in this paper, we only consider visual information.

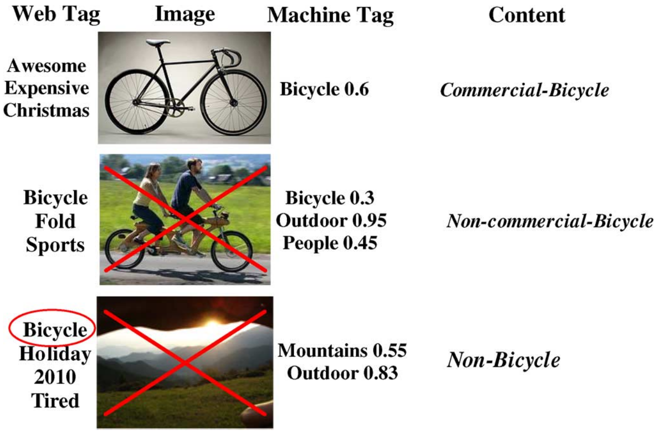


Fig. 1. Imagine that we want to buy a new bicycle and we are searching pictures of bicycle in a commercial context. Using user-contributed web tags or automatic machine tags in isolation will not be sufficient.

To improve the effectiveness of visual information, we could use the observation that large video collections in general contain different genres, such as news broadcast, home-video, advertisement, music, sports, and movies. Computing complex concept models for all of the data in every genre requires large computational resources. Having videos classified into genres is one way to make the search task more efficient. In many applications, the user is interested in one or two specific genres only. For example, imagine that we want to buy a new bicycle; we then need bicycle pictures from commercial advertisements, not from real-life photograph collections containing a bicycle, as illustrated in Fig. 1. More examples are a police investigator examining a hard drive of a computer only for illegal material, a new parent searching videos of baby products, and a sports fan exploring cars in formula one videos. When doing topic search, the topic “Barack Obama” or “Bush attacked using shoes during press conference in Iraq” only need to be searched within the “News-Broadcast” genre. Thus, to handle large-scale video sets, using genres might be a feasible and practical solution.

Semantic indexes to videos can be computed at two different levels, namely at the genre level related to the video as a whole and at the semantic concept level (or at the topic level in topic search) which operates on shot or subshot level. To detect genres, we can make use of the fact that a video genre is a set of videos sharing similar style [2], which is chosen by the director of the program, where the style fits the purpose of the genre. In automatic image annotation, Duan [3] tries to label images in groups taken at the same location, with the same setup, or over

Manuscript received December 27, 2010; revised October 08, 2011; accepted October 19, 2011. Date of publication November 07, 2011; date of current version March 21, 2012. This was supported in part by the European Union Project “The Investigator’s Dashboard (through the Safer Internet plus program SIP-2007-TP-1317033) and the National Science Foundation of China under Grant 61103062. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Ketan Mayer-Patel.

J. Wu was with the Intelligent Systems Lab Amsterdam (ISLA), Informatics Institute, University of Amsterdam, 1098 XG Amsterdam, The Netherlands. He is currently with the School of Electronics and Information, Northwestern Polytechnical University, Xi’an 710072, China (e-mail: junwu@nwpu.edu.cn).

M. Worring is with the Intelligent Systems Lab Amsterdam (ISLA), Informatics Institute, University of Amsterdam, 1098 XG Amsterdam, The Netherlands (e-mail: m.worring@uva.nl).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2011.2174969

the same trip. Automatic video genre categorization [4]–[6] is feasible when sufficient metadata are available, but, in cases where these are lacking, a content-based solution is required. At the semantic level, research in concept-based video indexing focuses on building large numbers of unrelated individual semantic concept detectors [7]–[9] such as *sunsets*, *indoor*, *outdoor*, *cityscape*, *landscape*, and *forests*, or creating a set of concept detectors based on knowledge such as the concept ontology described in [10]–[12].

Genres and semantic indexes have an intimate relation. When considering this relation, we should distinguish the following two cases. The first case occurs when a concept is specific to one or two genres, i.e., it almost never occurs in other genres. For example, *MovieStar*s appear in *Movies*, and *AnchorPerson* occurs in *Broadcast* only. As these concepts only appear in certain genres, we can focus on a genre-relevant subset, ignoring other irrelevant genres. In the second case, one concept might occur in several genres where it has different visual characteristics. For instance, considering the *Person* concept. In a movie, actors seldom look into the camera directly, whereas in home video or in mobile phone video talking heads or frontal faces appear quite often. Another example is the concept *Table* in the *Meeting* genre, which is more restricted than the generic *Table* concept. For many concepts, there are variations among different genres, resulting in diverse visual appearances for one concept. Obviously, this variation becomes less if we restrict the analysis to videos in a particular genre type, and hence concept detection becomes easier. When the users only care about a certain concept within a target genre (such as the *Commercial-Bicycle* example), the generic concept models are prone to be under-fitting in such a narrow domain. To improve, genre-specific concept models need to be derived from the subset within the specific genre.

In the above cases, it is possible to utilize video genre classification to filter out most of the irrelevant materials, resulting in a relatively small subset of the original data set, as illustrated in Fig. 2. Consider the above case of buying a bicycle again. Users are only interested in *Commercial-Bicycle*, so all of the *Non-commercial-Bicycle* and *Non-Bicycle* video material can be thrown away. Using such a method also contributes to efficiency when efficient global feature are applied for genre classification as this step works on the full set and subsequent steps on smaller subsets.

In this paper, we consider detecting genre-specific semantic concepts for a given video genre in an efficient way by creating a two-step framework. In the first stage, the video genre models are trained using efficient global features, and then the genre-specific concept models are trained using complex local object-level features, within the target video genre only. In the classification stage, we first perform video genre classification to quickly filter out most of the irrelevant videos. The genre-specific models are then used to classify the remaining video subset. Of course, this introduces the risk of throwing away videos containing the target concept, so we need to carefully handle the tradeoff between accuracy and efficiency.

The remainder of this paper is organized as follows. In Section II, we review related work. We introduce our framework in Section III. In Section IV, extensive experiments are presented, followed by conclusions in Section V.

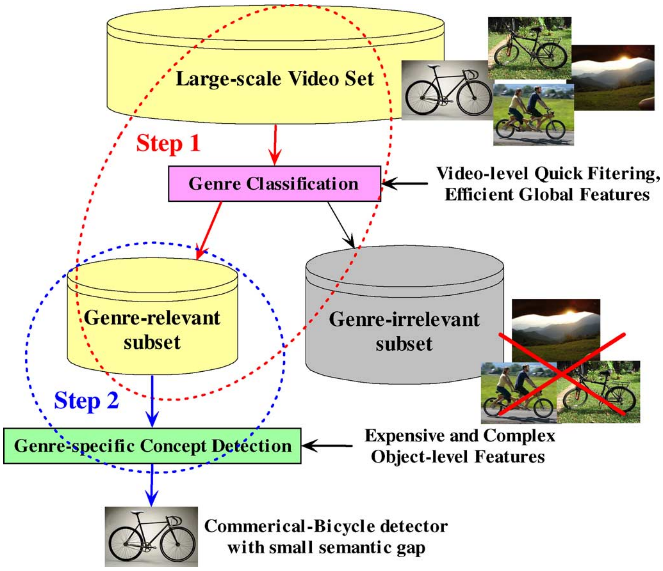


Fig. 2. Our two-step framework for genre-specific video indexing. First, based on efficient video genre classification, we reduce the entire set into a much smaller subset by performing video-level quick filtering to filter out most of the irrelevant videos. Then, the more complex genre-specific concept models using expensive object-level features are used to classify the estimated subset of the test set, based on the outputs of the video genre classification.

II. Related Work

We briefly introduce video genre classification and concept-based semantic indexing and then discuss how to detect genre-specific concepts.

A. Video Genre Classification

Automatic video genre classification started with Fisher [4] in 1995. It has been followed by many other works focusing on archive data [5], [13], [14] or web data [6], [15]–[17]. Most of these methods consider video genres as a one-layer flat structure, while a hierarchical ontology is applied in work such as [5] and [17].

By using title-based information only, Song [16] proposes an incremental support vector machine (SVM), with the help of online Wikipedia propagation, to categorize large-scale web videos. Wu [15] combines contextual and social information for web video categorization. The semantic meaning of text (title and tags), video relevance from related videos, and user interest are integrated to robustly determine the video genre. In [6], Yang proposed a multimodality web video categorization algorithm, which includes a semantic modality and a surrounding text modality. In Borth's TubeFiler framework [17], the first-level genre is found based on text information such as tag and title, and the second-level subgenres are fine-grained based on visual features. All of these methods rely on sufficient metadata such as texts or tags; when these metadata are not available at all, a content-based method is the only feasible solution.

Yuan [5] presents such a content-based solution based on a hierarchical ontology of video genres, in which the basic genres are: movie, commercial, news, music video, sports, and so on. Then, only two of those (movie and sports) are further divided into a couple of subclasses. Their hierarchical SVM is effective for the generic problem of video genre classification but ignores the fact that, in many applications, users might be only interested in a limited number of genres for further analysis. In these cases, treating each level in the same way (with the same model and the same feature) is not so efficient, especially for large-scale video sets. In addition, they do not make a connection from a genre to a concept within that genre.

B. Semantic Indexing

Until now, many concept detectors have been obtained using different pattern recognition techniques. In early literature [7]–[9], specific concept detectors, such as *news anchor person*, *sunsets*, *indoor*, *outdoor*, *mountains*, and *forests*, have been developed specifically for the target concepts. Other work explores the relationships between semantic concepts, such as hierarchical models [11], the co-occurrence of two concepts [18], actions, or objects in context [19], [20] and inter-concept relationships [12], [21], [22].

The TREC Video Retrieval Evaluation (TRECVID) conference [23], [24], started in 2001, provides a large test collection as a benchmark for all participants. Among others, systems such as IBM system [25], MediaMill from UvA [26], the Tsinghua System [27], the Columbia system [28], and the Informedia system from CMU [29] have been developed for this benchmark. With more and more powerful computing resources, detecting a large amount of concept detectors is now feasible [10], [30], [31]. TRECVID has started the trend to move from specific-purpose build models to generic models suited for every genre. This is a positive solution for generic concept detection, but methods require complex statistical models to cover the large variations in appearance over the different genres.

Our goal is different from the traditional concept detection as introduced above. Rather than having one complex model covering all genres, we consider a large set of simpler genre-specific models.

C. Genre-Specific Concept Detection

The methods described in the previous subsections either consider genre classification or concept classification. In [32], we introduced a method to detect genre-specific concept detection in the video domain by combining the genre and concept classification. This method yields a generic methodology for deriving genre-specific concept detectors. The fast document ranking [33] and Static Index Pruning [34] methods deal with similar issues in the IR domain. In the references, the genres are restricted to the results of a ranking where elements are possibly relevant and irrelevant; for video genre classification, we have to deal with diverse genres.

In this paper, we improve the work in [32] in three aspects, given here.

- Instead of using the same complex feature for both genre classification and genre-specific concept detection, we apply efficient global features in the stage of genre classification. Within a small subset, we then train genre-specific concept models using complex local object-level features.
- To obtain additional speed-up, we skip the step of shot segmentation and randomly extract a couple of frames from each video and show that the results remain at the same level of accuracy.
- In addition to the 28-h video set used in [32] for genre-specific concept detection, we use a large-scale benchmark set, which contains 80 031 YouTube videos totaling 4168 h, for genre-specific topic search.

III. Proposed Framework

We now detail our proposed framework, which is illustrated in Fig. 3. To predict a given genre-specific concept, we use a two-step strategy: video genre classification is first applied to filter out most of the irrelevant videos, and then the genre-specific concept models classify the samples within the estimated subset for the given video genre. The genre structure we deal with is not necessarily a hierarchy, as we are not targeting the general problem of genre classification. To detect the genre-specific concepts, in principle we only need to separate the whole data set into two subcollections: relevant and irrelevant.

Suppose there are in total J video genres $G = \{g_1, g_2 \cdots, g_J\}$, K semantic concepts $C = \{c_1, c_2, \cdots, c_K\}$, and let the genre-specific concept set be defined as

$$C_G = \{c_{1,1}, \cdots, c_{k,j}, \cdots, c_{K,J}\} \tag{1}$$

where $c_{k,j}$ is the concept $c_k \in C$ within the video genre $g_j \in G$.

Further, assume we have a training set with n videos $\mathbf{V_X} = \{v_1, v_2, , \cdots, v_n\}$, where all of the videos are segmented into video shots (small clips), and representative key-frames are extracted, resulting in a key-frame set $\mathbf{X} = \{x_1, x_2, \cdots, x_N\}$.

Our goal is to predict the posterior probability of a genre-specific concept, given a sample video shot s and not present in any of the videos in the training set V_X:

$$P(c_{k,j}|s) = \frac{P(s|c_{k,j})P(c_{k,j})}{\sum_{c_{k,j}} P(s|c_{k,j})P(c_{k,j})} \tag{2}$$

where $P(c_{k,j})$ is the prior probability of the genre-specific concept $c_{k,j}$. As some concepts are restricted to specific genres, many of these priors will be zero, and hence all of these posterior probabilities are also zero.

The detailed information of "video-level quick filtering" and "genre-specific concept detection" for both training and classification stages are now introduced step by step.

A. Two Schemes to Train Genre-Specific Concept Models

There are basically two ways to train genre-specific concept models. The first option is to train genre-and-concept models directly as classical methods; we call these *joint models*. The second option is to train concept models only within the target genre, where the video genre classification in the operational phase is used to quickly filter out irrelevant videos; we call these *cascade models*.

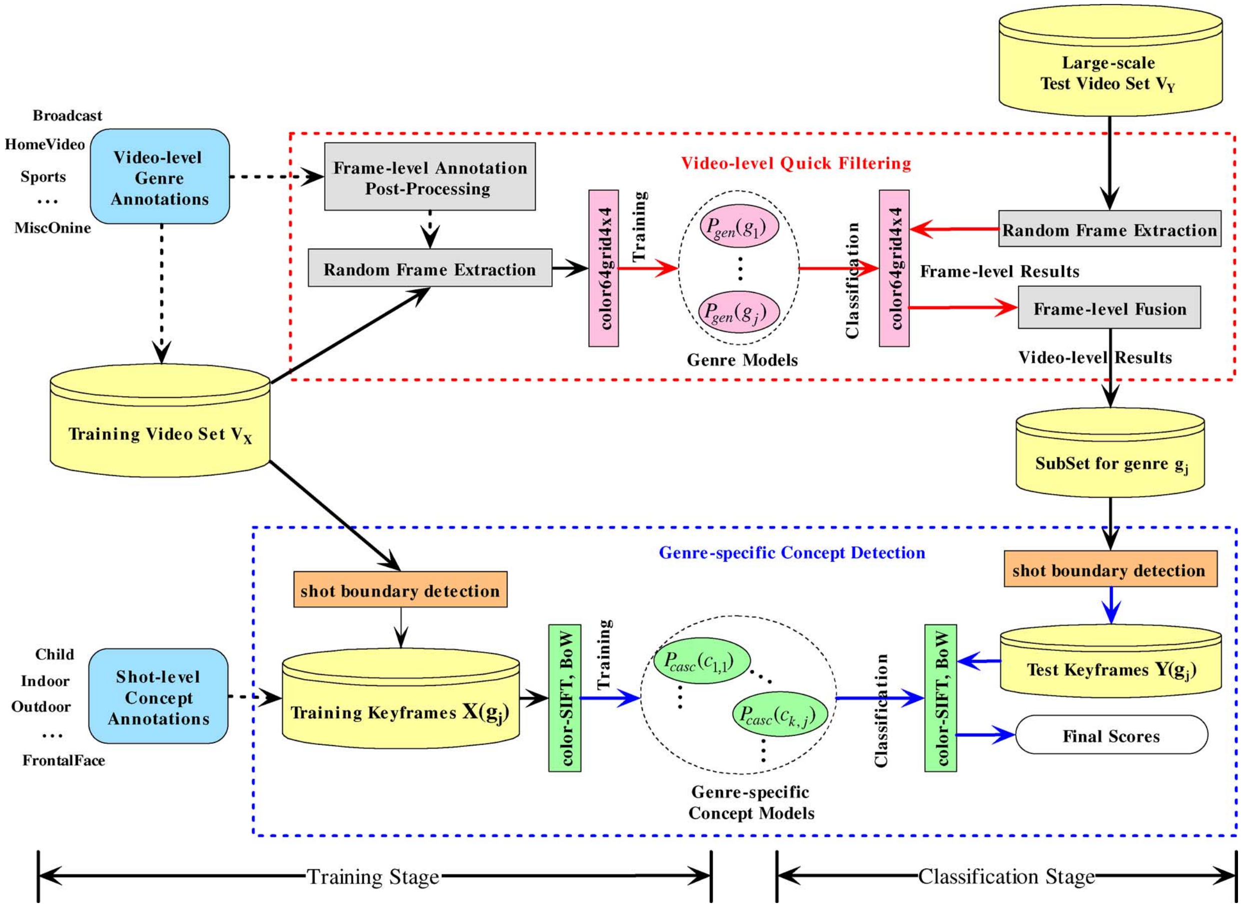
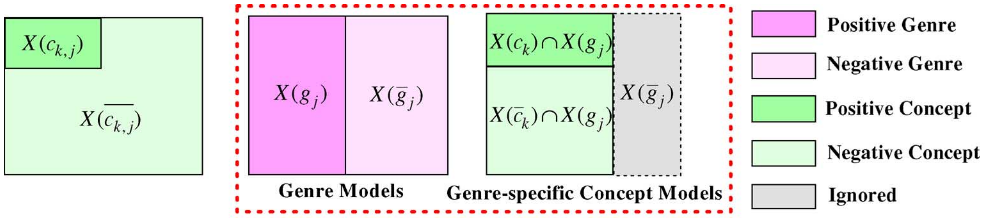


Fig. 3. Detailed data flow diagram of the proposed two-step framework (applying cascade models). If we ignore the step of video-level quick filtering, this diagram degenerates to the one for joint models.



(a) (b)

Fig. 4. Two schemes to train genre-specific concept models: (a) joint models and (b) cascade models. In each subfigure, the whole rectangle denotes all possible shots and their distribution for a genre g_j, a concept c_k, or a genre-specific concept $c_{k,j}$.

The required data annotations for the joint models and the cascade models are illustrated in Fig. 4. For a given target, which can be a video genre g_j, a genre-specific concept $c_{k,j}$, or combinations of c_k and g_j, let $X(\cdot)$ be a subset of the whole frame set $\mathbf{X}$ containing only positive samples of the given target and $X(\bar{\cdot})$ be a subset of the whole frame set $\mathbf{X}$ containing only negative samples of the given target. For example, for training joint models $P_{\text{join}}(c_{k,j})$, $X(c_{k,j})$ is the positive sample set, while $X(\overline{c_{k,j}})$ denotes the negative sample set. For training, the genre-specific concept model $P_{\text{casc}}(c_{k,j})$, $X(c_k) \cap X(g_j)$ denotes the positive sample set, and $X(\overline{c_k}) \cap X(g_j)$ indicates the negative sample set.

1) Joint Model: For the first option, we map the task of detecting the genre-specific concepts to a classical concept detection problem, but with different definitions of the classes. In particular, we define classes as the intersection of a genre and a concept. For each genre-specific concept $c_{k,j}$, the key-frames in each shot are labeled as positive (positive for genre and positive for concept) and negative otherwise. For a given genre-specific concept $c_{k,j}$, we build a model $P_{\text{join}}(c_{k,j})$ as

$$P_{\text{join}}(c_{k,j}) = P\left(c_{k,j}|\mathrm{X}(c_{k,j}), \mathrm{X}(\overline{c_{k,j}})\right) \qquad (3)$$

based on the annotations $\mathrm{X}(\cdot)$ for a genre-specific concept $c_{k,j}$.

Finally, we apply the joint model $P_{\text{join}}(c_{k,j})$ to retrieve the posterior probability of the given shot

$$P_{\text{join}}(c_{k,j}|s) = \frac{P_{\text{join}}(s|c_{k,j})P(c_{k,j})}{\sum_{c_{k,j}} P_{\text{join}}(s|c_{k,j})P(c_{k,j})}. \qquad (4)$$

2) Cascade Model: For the second option, we apply a two-step strategy: the genre models are trained based on video-level genre annotation, and the genre-specific concept models are trained (and applied) only within the target genre.

Now, for a given video genre g_j, we build a model

$$P_{\text{gen}}(g_j) = P\left(g_j | \mathrm{X}(g_j), \mathrm{X}(\overline{g_j})\right), \tag{5}$$

based on the set of training videos. With the genre model, given a sample video y, we can compute the probability

$$P_{\text{gen}}(g_j|y) = \frac{P_{\text{gen}}(y|g_j)P(g_j)}{\sum_{g_j} P_{\text{gen}}(y|g_j)P(g_j)}. \tag{6}$$

For a given genre-specific concept $c_{k,j}$, we build a model

$$P_{\text{casc}}(c_{k,j}) = P\left(c_{k,j}|X(c_k) \cap X(g_j), X(\overline{c_k}) \cap X(g_j)\right) \tag{7}$$

based on a set of annotated examples for $c_{k,j}$ only within the target genre g_j. The posterior probability of the given shot for cascade models are

$$P_{\text{casc}}(c_{k,j}|s) = \frac{P_{\text{casc}}(s|c_{k,j})P(c_{k,j})}{\sum_{c_{k,j}} P_{\text{casc}}(s|c_{k,j})P(c_{k,j})}. \tag{8}$$

For the cascade models, it is worth noting that they save a large amount of human labor during the training stage, as the video-level genre annotation is much quicker than shot-level concept annotation. The advantage of cascade models is that we can deal with genre classification more efficiently (using simple features) at the video level and handle the subsequent genre-specific concept detection (only on a subcollection) at the shot level.

B. Video-Level Quick Filtering

1) Shot-Level Genre Models: Video-level genre annotations do not indicate which individual video frames are specific to the target video genre and which do not. As a consequence, the genre type of each video frame can be derived from the master video clip, but with noise. For example, there might be black frames that are not suitable as positive samples and likely to mislead the training process of the shot-level genre models. In addition, duplicated frames within the same video are redundant and might bias the classification. To deal with this, all of the black frames are eliminated from both the training set and test set for genre classification. Duplicated frames are also eliminated from the training set, keeping only one instantiation.

It should be noted that, though the video-level genre annotation is more efficient than shot-based or frame-based genre annotation, this annotation strategy also has its side effects, including, in particular, noisy data annotation when considering the annotations individual frames receive. For a large-scale video data set, shot-based or frame-based annotation is almost not possible. Thus, the above-mentioned procedure for video-level annotation is the only feasible solution.

Based on the annotations of the J video genres, a set of genre models can be trained as

$$\mathbf{P}_G = \{P_{\text{gen}}(g_1), P_{\text{gen}}(g_2), \cdots, P_{\text{gen}}(g_J)\} \tag{9}$$

where $P_{\text{gen}}(g_j)$ is defined in (5).

As the genre classification has to be performed on the whole collection, for large-scale video data, the step of shot segmentation is still expensive. To improve efficiency, we observe that actually this segmentation step might be skipped completely, and a set of representative frames $\mathbf{X}' = \{x'_1, x'_2, \cdots, x'_N\}$ (instead of the former key-frame set $\mathbf{X} = \{x_1, x_2, \cdots, x_N\}$) for a video might be sufficient. Thus, we perform *random frame extraction* from both the training and test videos for the quick filtering step. Further, we apply efficient global features such as color-only features to obtain the shot-level video genre models.

2) Video-Level Fusion: The shot-level genre model $P_{\text{gen}}(g_j)$ outputs the shot-based score $P(g_j|s)$ for genre g_j. To make the classification more precise, these shot-level scores within a given test video y are combined into a video-level score by

$$P_{\text{gen}}(g_j|y) = \frac{1}{N(y)} \sum_{s \in y} P_{\text{gen}}(g_j|s) \tag{10}$$

where $N(y)$ is the number of shots within the test video y.

Suppose V_Y is the test video set. For the target genre g_j, we discard *useless* videos (i.e., videos more likely to be in other genres) according to their scores. These scores are computed based on posterior probabilities of the target video genre, and only those with scores higher than a predefined threshold are kept for the final concept classification stage. What remains is a relatively small subset of the test set

$$\tilde{\mathrm{V}}_Y(g_j, \gamma) = \{y : P_{\text{gen}}(g_j|y) > \gamma\} \tag{11}$$

which will be the input for the genre-specific concept classification in Section II-C, with γ a threshold parameter.

C. Genre-Specific Concept Detection

1) Training Genre-Specific Concept Models: For each genre g_j, we train a set of genre-specific concept models within all of the videos from the genre g_j, i.e., all of the videos outside the target genre will be ignored. The genre-specific concept models are

$$\mathrm{P}^S(g_j) = \{P_{\text{casc}}(c_{1,j}), P_{\text{casc}}(c_{2,j}), \cdots, P_{\text{casc}}(c_{k,j})\} \tag{12}$$

where $P_{\text{casc}}(c_{k,j})$ is defined in (7). Accordingly, the full set of genre-specific concept models is

$$\mathbf{P}^S(G) = \left\{\mathrm{P}^S(g_1), \mathrm{P}^S(g_2), \cdots, \mathrm{P}^S(g_J)\right\}. \tag{13}$$

2) Applying Genre-Specific Concept Models: Based on the above estimated subset $\tilde{\mathrm{V}}_Y(g_j, \gamma)$, we apply the genre-specific model $P_{\text{casc}}(c_{k,j})$ to obtain the posterior probability of the sample test shot belonging to the genre-specific concept $c_{k,j}$, $P_{\text{casc}}(c_{k,j}|s)$. To make the search more efficient, the subcollection of videos to be searched is itself ordered by decreasing the likelihood of containing those concepts. This further improve the efficiency and effectiveness.

We summarize all of the above models in Table I. For the cascade models as an example, when applying video-level genre

TABLE I
POSTERIOR PROBABILITIES OF DIFFERENT MODELS: 1) JOINT MODELS AND 2) CASCADE MODELS

Name	Model	Category
joint model	$P_{join}(c_{k,j}\,|\,s)$	1) joint models
genre model	$P_{gen}(g_j\,|\,y)$	2) cascade models
genre-specific	$P_{casc}(c_{k,j}\,|\,s)$	2) cascade models

models on the test set V_Y, we get a series of posterior probabilities $P_{\rm gen}(\cdot)$ given a test sample video y from the test set. Based on these scores, the irrelevant videos will be eliminated from the test set for further processing. Based on the remaining subset, a ranking of the input shots is obtained based on genre-specific concept or topic models.

Our two-step framework has the additional advantage that we might not have to use complex features for the ultimate application of concept detection (depending on the target concept itself) because we divide the data to be processed into groups of more specific concepts which can be expected to have less variation in their visual appearance.

IV. EXPERIMENTS

We now experimentally verify the effectiveness and efficiency of our two-step framework in genre-specific concept detection and in genre-specific topic search. We first show the performance of video genre classification, and then we evaluate the genre-specific models with both ground-truth genres and estimated genres. For evaluating the effectiveness, we compare the cascade models with the joint models. For evaluating efficiency, we consider how much time can be saved when applying the two-step cascade models compared with the case of using complex object-level features and at what loss in performance.

A. Data Set and Basic Setups

1) Data Set: We use two data sets to evaluate our method for two different tasks. The first set is a large benchmark set (with incomplete topic annotation available), which is used to evaluate genre-specific topic search. While the second smaller 28-h dataset (with complete concept annotation available) is utilized to evaluate the genre-specific concept detection.

The first set is the MCG-WEBV benchmark set [35] from the Chinese Academy of Sciences, which contains 80 031 YouTube videos from December 2008 to February 2009. This comprises the most viewed videos of every month on YouTube and is called the "core set." The database of the related videos and ones uploaded by the same authors is called the "expanded set". As listed in Table II, the core set has 3283 videos, and the expanded set contains 76 748 videos. For video genre classification, we use one month's video in the core set as the training set (mcgcoredevel), another two months of videos in the core set as the validation set (one month for mcgcoretest1a) and the test set (one month for mcgcoretest1b), respectively. For the genre-specific topic search, the whole core set (mcgcore) is applied as the training set, and the expanded set (mcgexpanded) is used as the test set. In total, 15 YouTube video genres have already been labelled based on category tags on the whole data set. Based on a topic cluster strategy [35], in total 73 topics (see Table III for ten topics with more positive videos) are manually labelled only on the core set. Note that the topic annotation on the expanded set is not available. We use these incomplete annotations (not all positive videos are annotated) for training the genre-specific topic models in our framework.

TABLE II
MCG-WEBV BENCHMARK SET (VERSION 1.0) CONTAINS TWO MAIN SUBSETS. THE CORE SET CONTAINS 3283 VIDEOS, AND THE EXPANDED SET CONTAINS 76 748 VIDEOS. THE CORRESPONDING UPLOADING TIME IS FROM DECEMBER 2008 TO FEBRUARY 2009

Subset	#Videos	#Key-frames	Months
mcgcoredevel	1,161	30,756	12/2008
mcgcoretest1a	1,072	28,643	01/2009
mcgcoretest1b	1,050	39,907	02/2009
mcgcore	3,283	99,356	12/2008–02/2009
mcgexpanded	76,748	2,084,581	12/2008–02/2009

Another small 28-hour video set, six video genres ($g_1 \sim g_6$: *Broadcast*, *HomeVideo*, *Porn*, *MiscOnline*,[1] *Sports*, and *Soap*) are manually labelled at video level. Based on the above genre annotations, seven semantic concepts within each genre are annotated at shot level ($c_1 \sim c_7$): *FrontalFace*, *Child*, *Indoor*, *Outdoor*, *CloseupNudity*, *CloseupBreast*, and *Women-Bikini*. In the use case mentioned above (a police investigator examining a hard drive of a computer for illegal material), the police investigators are interested in specific concepts such as *CloseupNudity* and *CloseupBreast*. Furthermore, the people related concepts such as *child* and *CloseupNudity* and the scene concepts such as *Indoor* and *Outdoor* provide important clues for further investigation. We detect 15 genre-specific concepts (formatted as genre-concept): *Broadcast-FrontalFace*, *Broadcast-Indoor*, *Broadcast-Outdoor*, *HomeVideo-Child*, *HomeVideo-Indoor*, *HomeVideo-Outdoor*, *MiscOnline-Indoor*, *MiscOnline-Outdoor*, *MiscOnline-Women-Bikini*, *Porn-CloseupNudity*, *Porn-Indoor*, *Soap-FrontalFace*, *Soap-Indoor*, *Sports-Indoor*, and *Sports-Outdoor*.

2) Experimental Setup: For a certain genre-specific concept $c_{k,j}$, each video within the target genre is first segmented into video shots, and representative key-frames are selected from each shot. Thus, for the target genre g_j and for each genre-specific concept, all the key-frames in each shot are labelled as positive or negative samples. For the genre classification, we extract simple color-only features denoted by color64 (44-dim color correlogram [36], 14-dim color texture moment [37], and 6-dim RGB color moment), and grid-based color64 features (2 × 2, 3 × 3, 4 × 4, etc.).

For the black frame filtering, we construct a 16-bin histogram in gray scale, and filter out any frame with the value of the first bin smaller than a predefined threshold (0.99 in our experiments). For the duplicate image detection, we use a similar strategy as in [38]. The grid2 × 2-based color64 feature is first reduced by Principle Component Analysis, and then a 32-bit hash code is computed as the unique identity of an input image. Two images sharing the same hash code are considered as duplicate images. Under this setting, both the black frames

[1] *MiscOnline* means the whole data set excluding the other five video genres. This rest set mainly includes some small videos downloaded from the Internet.

TABLE III
TEN (OUT OF 73) TOPICS IN THE MCG-WEBV BENCHMARK DATA SET. WE ONLY SELECT THE TOPICS HAVING 20 POSITIVES OR MORE

ID	#Videos	Description
1	32	Bush was attacked by shoes during press conference in Iraq
3	20	New characteristic in google earth 5.0.
6	27	News: this topic is about an event that US Airways 1549 crashes in Hudson River near Manhattan.
14	22	Barack Obama: this topic includes address, inauguration, dance, music video sing and impersonator about Barack Obama.
18	24	The Philip Defranco Show: this topic is about a blogTV show from Philip Defranco.
33	21	A topic about a popular online gaming world of warcraft (WOW in short), including the operate guides and ads.
34	36	Teaching you how to make money online
35	27	Naruto Manga: Naruto is a full-length cartoon series created by Japan Kishimodo Masashi
46	23	Makeup introduction: this topic talks about how to do face makeups, with their emphasis on eye makeup.
48	34	Funny cats : picture collections and videos of very funny and cute little kitten.

(around 1%) and the duplicated frames (around 11%) are eliminated from the data set.

For the genre-specific concept detection or genre-specific topic search, the dense sampling detector [39] and Harris-Laplace salient point detector are applied in the opponent color space. Then, OpponentSIFT [40] features, a variant of SIFT [41], are extracted based on a spatial pyramid with a 1 × 1, 2 × 2, and 1 × 3 layout. On top of that, the Bag of Words model [42] is employed.

The SVM with a χ^2 kernel [43], [44] is used for learning. Parameter tuning for the SVM is conducted on the training set using a threefold cross validation. The tuning of the parameter γ in (11) is conducted on the validation set beforehand.

B. Evaluation Criteria

To evaluate different aspects of video genre classification, genre-specific concept detection and genre-specific topic search, we apply evaluation criteria as follows.

1) Video Genre Classification: To evaluate the result of video genre classification, we use a *confusion matrix* to see which genres are confused most.

2) Genre-Specific Concept Detection: We use the *Average Precision* (AP) as a measure to evaluate the shot-level results of the video genre classification in the top 500 returned video shots (AP@500) and in the top 1000 returned video shots (AP@1000).

The *efficiency curve* is defined as the relative loss in performance [e.g., measured as mean average precision (MAP)] versus the percentage from the test data set kept after filtering. The *relative* loss is compared with the case where the full data set is kept and normally the highest MAP is achieved. When only part of the test videos are kept, the performance is likely to have a certain loss. The efficiency curve illustrates the trend of this performance loss based on different subsets with different percentages kept.

3) Genre-Specific Topic Search: The *precision* in the top 20 items (P@20) is important in web-based search. Due to the absence of topic annotation in the test set, we use this measure to evaluate the genre-specific topic search.

C. Results of Video Genre Classification

1) Video-Level Genre Results: We illustrate the confusion matrix of the genre classification at video level for 15 YouTube genres (the 15 common YouTube channels from www.youtube.com) in the benchmark YouTube video set in Fig. 5. We observe that for several genres a quite reasonable performance is obtained while for others especially for genres containing many people, such as *Comedy*, *Entertainment*, *Peple-Blogs*, and *Travel-Events* performance is limited.

2) Feature Evaluation: In a small data set, the Opponent-SIFT feature can be applied for genre classification. However, for a large data set, it is not feasible to use such expensive features. To use an as-simple-as-possible feature, we evaluate ten different features to see which feature can achieve reasonable performance while keeping processing speed at appropriate level. The results of the video genre classification of these ten features are illustrated in Fig. 6. The best performance is the OponentSIFT feature. The performance of grid4 × 4-based color64 decreases about 5.6%, while the processing speed is hundreds of times faster than the OponentSIFT feature (see more details in Section IV-F3). We will use this color64 feature for the step of video genre classification.

3) Random Frame Selection Versus Segment-Based: We compare the results of video genre classification using random frame selection versus a segment-based strategy in the 28-h video set, as in Table IV. For random frame selection, we divide each video file into equal-length video clips containing 100 continuous frames and randomly extract one representative frame from these small video clips. The F1 measure ($F1 = 2p * r/(p + r)$, where p denotes precision and r denotes recall), is used for evaluation. From this table, we can see that video-genre classification using more efficient random frame selection can achieve comparative performance in comparison to a segment-based strategy.

D. Results of Genre-Specific Concept Detection

1) Cascade Model Evaluation: As mentioned above, the two-step cascade models save large amount of human labors for annotating training data. They also speed up the semantic indexing process by reducing the original data set into a relatively small genre-specific subset. In this evaluation, we will show how much time can be saved, and at what loss in performance.

The efficiency curve of the cascade model is illustrated in Fig. 7. When keeping the full test set (100%), the (baseline) MAP is 0.478 (the loss is zero in this case). When using the two-step cascade models to filter out 50% of the data, the (relative)

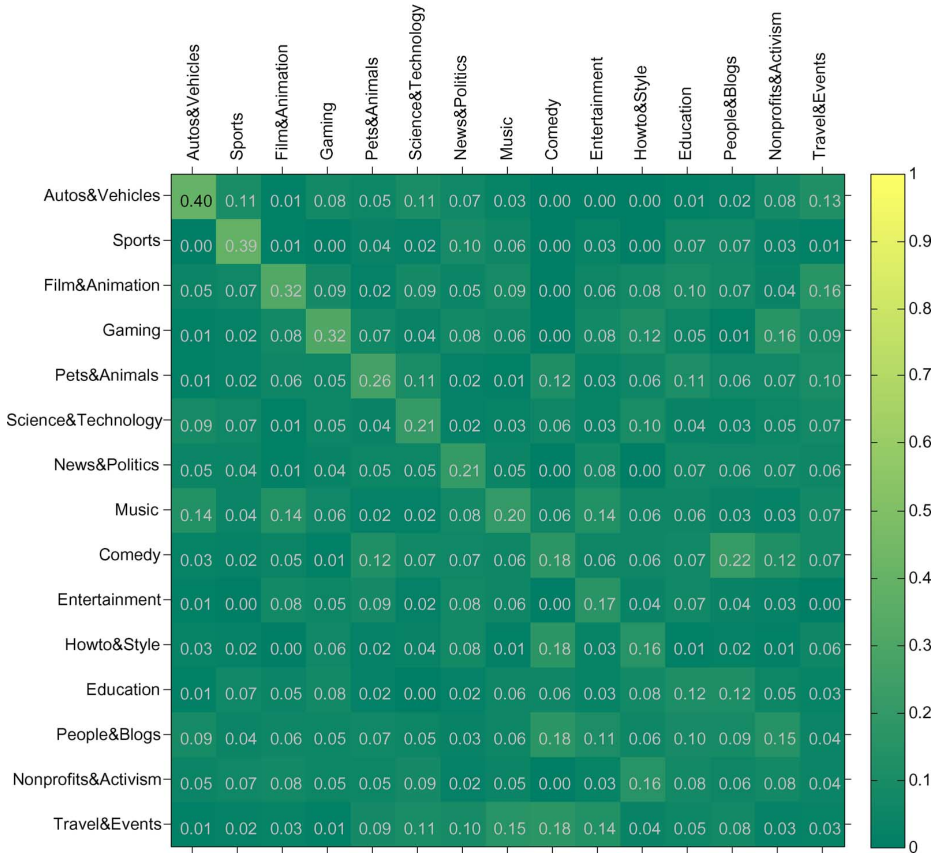


Fig. 5. Confusion matrix of the video-based genre classification results, where the diagonal values are used to sort the x/y-axis entries.

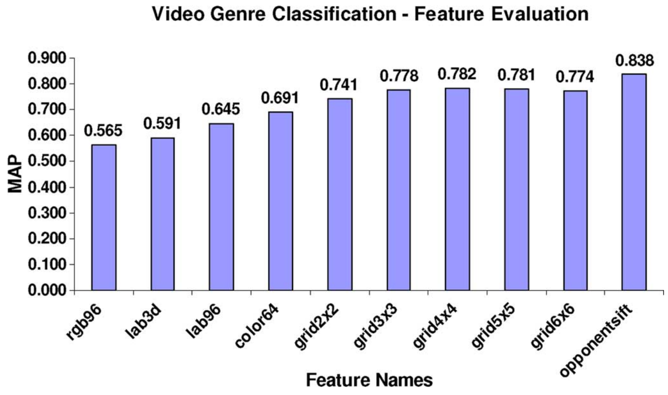


Fig. 6. Feature evaluation based on the task of video genre classification. These ten features are rgb96: 96-dim RGB-space histogram, lab3d: 150-dim $6 \times 5 \times 5$-cubic LAB-space histogram, lab96: 96-dim LAB-space histogram, color64: 64-dim color-only feature as in Section IV-A2, grid$\mathrm{N} \times \mathrm{N}$: $\mathrm{N} \times \mathrm{N}$ grid-based color64 feature, and oppenentsift: Opponent-SIFT feature.

TABLE IV

Comparison of Video-Genre Classification Using Random Frame Selection versus Segment-Based Strategy. The F1 Measure is Computed Based on Precision and Recall at the Video Level

Genre	Prior	F1-segment	F1-random
Broadcast	25.7%	0.687	0.667
HomeVideo	13.9%	0.923	0.892
Porn	11.7%	0.895	0.878
MiscOnline	48.8%	0.951	0.957
Sports	9.4%	0.727	0.720
Soap	16.3%	0.571	0.571

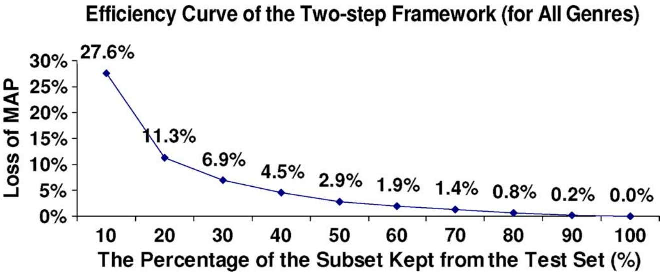


Fig. 7. The overall efficiency curve of the cascade models. The x axis denotes the percentage of the test data set kept, while the y axis denotes the loss in the Mean Average Precision (MAP). We consider 15 genre-specific concepts in all the six video genres.

loss of MAP is about 2.9% (0.014 in absolute loss, $2.9\% = (0.478 - 0.464)/0.478 = 0.014/0.478$). When 20% of the data are kept for further processing, i.e., 80% of the data are ignored, the relative loss of MAP is about 11.3% (0.054 absolute loss, $11.3\% = (0.478 - 0.424)/0.478 = 0.054/0.478$). Even when 90% of the data are ignored, the relative loss of MAP is only 27.6%.

Fig. 7 shows the results for all of the six genres together. However, the prior probabilities for different video genres vary. Thus, we show similar efficiency curves for each video genre separately in Fig. 8. From the figure, we observe that, if we keep approximately a percentage of the data equal to its prior probability for each video genre, the average performance for these

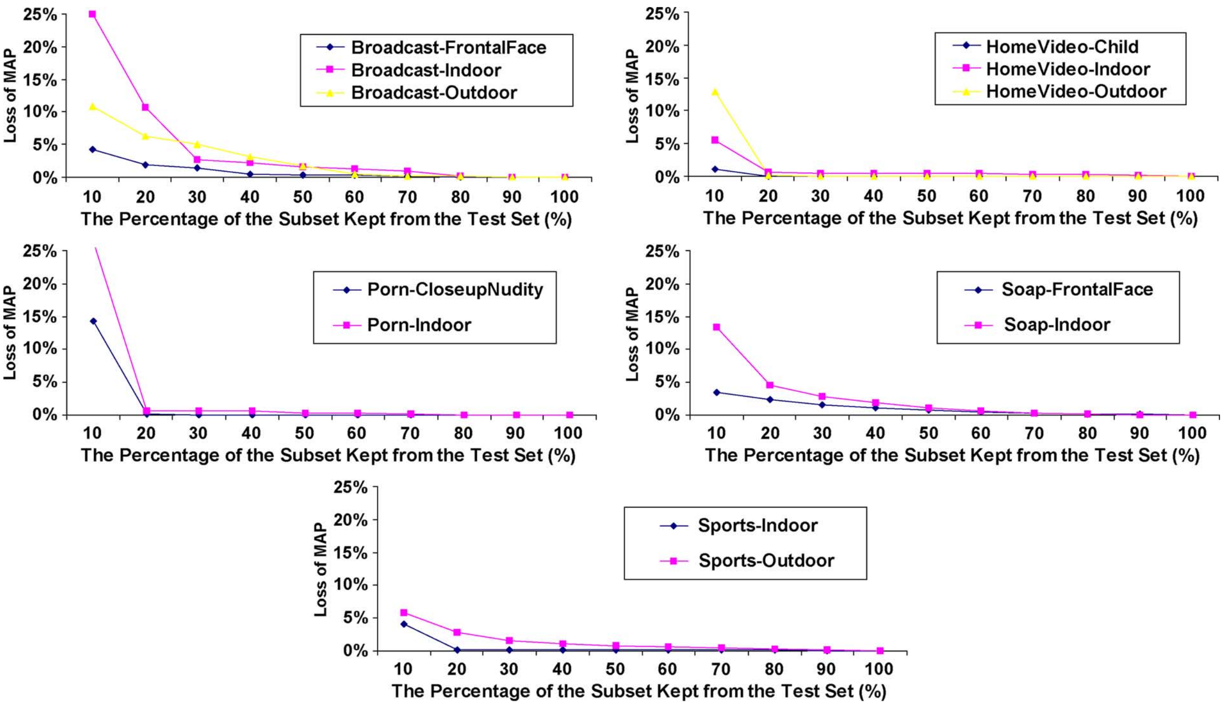


Fig. 8. Genre-specific efficiency curves of the cascade models. The x-axis denotes which percentage of the test data set is kept; the y-axis denotes the loss in MAP of the genre-specific concept models. Different from Fig. 7, we consider each video genre separately, resulting in five subfigures. These subfigures show some representative results for different video genres.

six genres decreases about 12%. If we keep a large portion of the data, e.g., twice its prior probability for each genre, the average performance decreases about 2%. Consequently, we conclude that our two-step framework can easily throw away most of the useless materials for the genre-specific concepts in each target genre, while the loss in performance is in a reasonable range.

2) Scheme Comparison: We compare our proposed two-step framework (cascade models, A1) with the classical concept detection method (joint models, B). In addition, to get a better understanding of our framework, we also use the ground truth of the video genres to obtain a subset for each target genre. We denote this variant of cascade models as scheme A2, defining the theoretical limit in performance.

The performance of the above two schemes is listed in Fig. 9.

We conclude that, with accurate information of the video genres, a variant of the cascade models (A2) indicates the upper bound of the performance for the original cascade models (A1) using the results of the video genre classification. The cascade models (A1) are consistently better than the parallel models (C) and the joint models (B) and perform rather close to the theoretical limit.

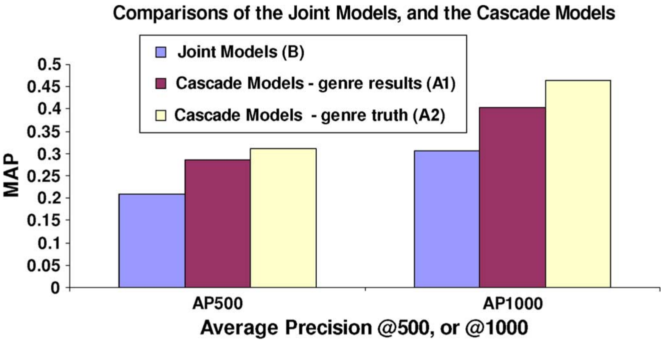


Fig. 9. Comparing the cascade models with the joint models and a variant of cascade models using the genres' ground truth (instead of using the predicted genre results). The MAP in the first 500 (left group) and 1000 (right group) shots are listed.

E. Results of Genre-Specific Topic Search

We execute the task of genre-specific topic search on the MCG-WEBV data set. The provided annotations (only positive videos for each topic) for the 10 topics (see Table III) in the training set (mcgcore) are double-checked, and wrongly-annotated videos are eliminated from these topic annotations. Accordingly, these 10 genre-specific topics are: *News&Politics-1* (Bush was attacked), *Howto&Style-3* (Google Earth 5.0), *News&Politics-6* (US airways crash), *News&Politics-14* (Barack Obama), *Entertainment-18* (Philip Defranco), *Gaming-33* (WOW game), *Howto&Style-34* (Making money online), *Film&Animation-35* (Naruto Manga Cartoon), *Howto&Style-46* (Makeup introduction), and *Pets&Animals-48* (Funny cats).

The efficiency curves of genre-specific topic search are illustrated in Fig. 10. These results are similar to the results of genre-specific concept detection. The main difference is that P@20 is more sensitive than MAP, as one missed positive shot decreases P@20 by 5%.

Further, the average efficiency curves of genre-specific topic search are illustrated in Fig. 11. When keeping the full test set (100%), the (baseline) average P@20 is 64% (the loss is zero in this case). When using the two-step cascade models to filter

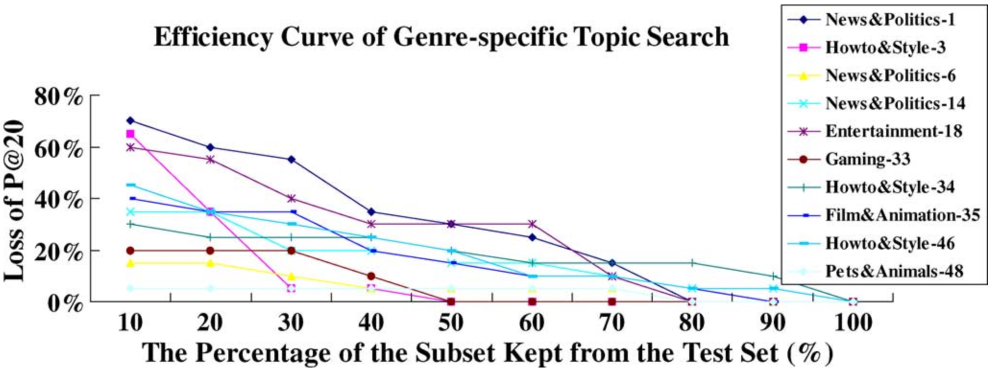


Fig. 10. Efficiency curve of the genre-specific topic search. The genre classification uses the $color64grid4 \times 4$ feature, and the genre-specific topic search utilizes the *OpponentSIFT* feature.

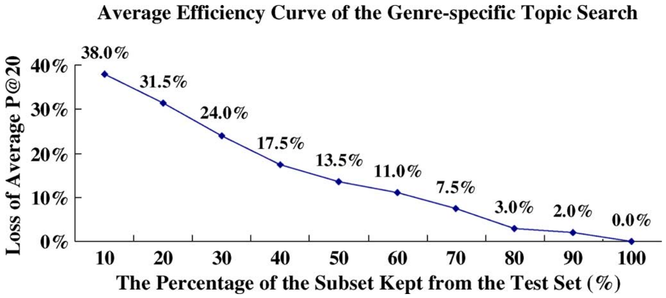


Fig. 11. Average efficiency curve of the genre-specific topic search. The performance loss is calculated based on average P@20.

out 50% of the data, the (relative) loss of P@20 is about 13.5% (0.135 in absolute loss). When 20% of the data are kept for further processing, i.e., 80% of the data are ignored, the loss of P@20 is about 31.5% (0.315 absolute loss). Even when 90% of the data are ignored, the loss of P@20 is less than 40%.

F. Speed-Up Factor

We compute the so-called *Speed-up Factor* (SF) as follows. For simplicity, we assume each video has the same length and the same resolution. We further assume that approximately the same number of video shots are found in a single video each having one key frame per shot. Finally we assume every genre has the same prior probability, $p_{g_j} = 1/J$. Let T be the processing time for the process.

1) Video-Level Genre Annotation: For the joint models, the genre-specific concept annotations are all based on shot-level, while in our two-step framework all the genre information are annotated based on video-level, and the genre-specific concept annotations are only executed at shot-level on a small subset. From the viewpoint of data annotations, the speed-up factor can be computed as

$$\mathrm{SF}_1=\frac{T_{An}(K\cdot J\cdot N)}{T_{An}(J\cdot n+K\cdot N\cdot p_{g_j})}=\frac{NKJ^2}{nJ^2+NK}=\frac{\frac{N}{n}KJ^2}{J^2+\frac{N}{n}K} \qquad (14)$$

where N is the total number of the video shots in the training video set V_X which contains n videos. Supposing $N/n = 100$, $K = J = 10$, $\mathrm{SF}_1 = 10^3/11 \approx 91$.

2) Video-Level Quick Filtering: After the step of genre classification, the videos that do not belong to the target genre g_j are ignored, resulting in a reduced test subset, that is, the processing time is reduced significantly, as

$$\mathrm{SF}_2(g_j) = \frac{1}{p_{g_j}(V_Y)} \approx J \qquad (15)$$

where $p_{g_j}(V_Y)$ is the prior probability of video genre g_j.

3) Using Efficient Features: For video genre classification, instead of using expensive features such as Opponent-SIFT, using efficient features such as $\mathrm{color64grid4} \times 4$ might be sufficient to achieve comparable classification performances. If we apply efficient features in the step of quick filtering, we can achieve further speed-up compared with the use of expensive complex features, as

$$\mathrm{SF}_3 = \frac{T(F_c)}{T(F_s)} \qquad (16)$$

where F_c is a complex feature such as SIFT or SIFT-like features and F_s is an efficient feature such as color-only features. We evaluate the Opponent-SIFT [40] (as F_c) and $\mathrm{grid4} \times 4$-based color64 features (as F_s) for video frames with 320×240 pixels on a dual-core of 3.00 GHz Intel Core2 Duo E8400 processor. The Opponent-SIFT feature takes about 1.5 s per frame (the time of creating the visual vocabulary is not considered here), the $\mathrm{color64grid4} \times 4$ takes about 4 ms per frame. In this case, the speed-up factor $\mathrm{SF}_3 \approx 375$.

Uijlings [45] recently reported an efficient DURF-based Bag-of-Words feature, approximating SIFT features, which can be extracted within 15 ms per image (ignoring the time of constructing visual vocabulary). This feature could be an interesting alternative for our simple color-based video genre classification, especially for genres containing many objects. In this case, the speed-up factor for extraction will be less, but likely it would allow for leaving out an even larger percentage of data when filtering. After that, full SIFT features could be applied.

4) Overall Speedup Factor: To simplify the comparison to the traditional methods of semantic indexing, we only consider the training process and classification process. The execution time of the traditional semantic indexing is $T_0 = T_{\mathrm{concept}}(\mathrm{set})$, and the executing time of our proposed two-level framework is $T_1 = T_{\mathrm{genre}}(\mathrm{set}) + T_{\mathrm{concept}}(\mathrm{subset})$. As in the training and classification of video genre it is possible to use efficient features, $T_{\mathrm{genre}}(\mathrm{set}) \ll T_{\mathrm{concept}}(\mathrm{subset})$, and we ignore the time of $T_{\text{genre}}(\text{set})$ when $\text{SF}_3 > J^2 = 100$ Finally, we combine these speed-up factors into an overall speed-up factor:

$$\text{SF} = \frac{T_0}{T_1} \approx \frac{T_{\text{concept}}(\text{set})}{T_{\text{concept}}(\text{subset})} = J \approx \text{SF}_2(g_j). \qquad (17)$$

In addition to the above, there is another speed-up in terms of the classification stage. The number of support vectors in the model will be less in general when a genre-specific model has been trained. For example, in the MCG-WEBV data set, the genre-specific topic SVM models have 470 support vectors on average, while the generic topic SVM models on average have 1551 support vectors (about 68% reduction). Hence, also the classification stage improves in efficiency.

V. Conclusion

In this paper, we propose a two-step framework to do genre-specific concept detection and genre-specific topic search. Models are built for each genre-concept or genre-topic combination where variation in appearance for each pair is less than for the general concept.

In the operational phase, video-genre classification is applied first to filter out most of the irrelevant material, resulting in a relatively small subset. Then, the genre-specific concept or topic models are applied. Experimental results show that our two-step method is efficient and more effective than joint models where the genre-concept examples are used in a regular supervised learning procedure. We show that, when filtering out 80% of the data set, the average performance loss is about 11.3% for genre-specific concept detection and 31.5% for genre-specific topic search, while the processing speed is hundreds of times faster depending on the video genre. As a result, we conclude that our two-step framework provides an efficient way to do genre-specific concepts detection and topic search. It is especially applicable in two conditions. When the importance of generic concepts over the genres is not balanced and are mostly to be found in specific ones, or when time is premium and the efforts should focus first on the most promising genres where in the background the less important concepts in other genres can be indexed.

Moreover, we can further improve the effectiveness of our proposed system by carefully determining the video genres. As our goal is not to find a solution for general genre classification (possibly hierarchically structured), the target genre-specific concepts in general only occur in one or two video genres. Thus, in our framework, in order to detect genre-specific concepts in the quick filtering step, we only need to distinguish two video genres at most. This will achieve better performance in genre classification than for 15 video genres or more.

References

[1] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*, 1st ed. Harlow, U.K.: Addison Wesley, 1999.

[2] C. G. M. Snoek and M. Worring, "Multimodal video indexing: A review of the state-of-the-art," *Multimedia Tools Applic.*, vol. 25, no. 1, pp. 5–35, 2005.

[3] A. Ulges, M. Worring, and T. Breuel, "Learning visual contexts for image annotation from flickr groups," *IEEE Trans. Multimedia*, vol. 13, no. 2, pp. 330–341, Apr. 2011.

[4] S. Fischer, R. Lienhart, and W. Effelsberg, "Automatic recognition of film genres," in *Proc. ACM Multimedia*, 1995, pp. 295–304.

[5] X. Yuan, W. Lai, T. Mei, X. sheng Hua, X. qing Wu, and S. Li, "Automatic video genre categorization using hierarchical SVM," in *Proc. ICIP*, 2006, pp. 2905–2908.

[6] L. Yang, J. Liu, X. Yang, and X.-S. Hua, "Multi-modality web video categorization," in *Proc. MIR*, 2007, pp. 265–274.

[7] J. R. Smith and S.-F. Chang, "Visually searching the web for content," *IEEE MultiMedia*, vol. 4, no. 3, pp. 12–20, Jul.–Sep. 1997.

[8] A. Vailaya, A. K. Jain, and H.-J. Zhang, "On image classification: City images versus landscapes," *Pattern Recognit.*, vol. 31, no. 12, pp. 1921–1936, 1998.

[9] A. Vailaya, M. A. T. Figueiredo, A. K. Jain, and H.-J. Zhang, "Image classification for content-based indexing," *IEEE Trans. Image Process.*, vol. 10, no. 1, pp. 117–130, Jan. 2001.

[10] M. Naphade, J. R. Smith, J. Tesic, S.-F. Chang, W. Hsu, L. Kennedy, A. Hauptmann, and J. Curtis, "Large-scale concept ontology for multimedia," *IEEE MultiMedia*, vol. 13, no. 3, pp. 86–91, Jul.–Sep 2006.

[11] J. Fan, H. Luo, Y. Gao, and R. Jain, "Incorporating concept ontology for hierarchical video classification, annotation, and visualization," *IEEE Trans. Multimedia*, vol. 9, no. 5, pp. 939–957, Oct. 2007.

[12] A. P. Natsev, A. Haubold, J. Tešić, L. Xie, and R. Yan, "Semantic concept-based query expansion and re-ranking for multimedia retrieval," in *Trans. ACM Multimedia*, 2007, pp. 991–1000.

[13] Z. Rasheed, Y. Sheikh, and M. Shah, "On the use of computable features for film classification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 1, pp. 52–64, Jan. 2005.

[14] L.-Q. Xu and Y. Li, "Video classification using spatial-temporal features and PCA," in *Proc. ICME*, 2003, pp. 485–488.

[15] X. Wu, W.-L. Zhao, and C.-W. Ngo, "Towards google challenge: Combining contextual and social information for web video categorization," in *Proc. ACM Multimedia*, 2009, pp. 1109–1110.

[16] Y. Song, Y.-D. Zhang, X. Zhang, J. Cao, and J.-T. Li, "Google challenge: Incremental-learning for web video categorization on robust semantic feature space," in *Proc. ACM Multimedia*, 2009, pp. 1113–1114.

[17] D. Borth, J. Hees, M. Koch, A. Ulges, C. Schulze, T. Breuel, and R. Paredes, "Tubefiler: An automatic web video categorizer," in *Proc. ACM Multimedia*, 2009, pp. 1111–1112.

[18] M. R. Naphade, I. Kozintsev, and T. Huang, "Probabilistic semantic video indexing," in *Proc. NIPS*, 2000, pp. 967–973.

[19] M. Marszałek, I. Laptev, and C. Schmid, "Actions in context," in *Proc. CVPR*, 2009, pp. 2929–2936.

[20] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie, "Objects in context," in *Proc. ICCV*, 2007, pp. 1–8.

[21] X.-Y. Wei, Y.-G. Jiang, and C.-W. Ngo, "Exploring inter-concept relationship with context space for semantic video indexing," in *Proc. CIVR*, 2009, pp. 1–8.

[22] Y.-G. Jiang, J. Wang, S.-F. Chang1, and C.-W. Ngo, "Domain adaptive semantic diffusion for large scale context-based video annotation," in *Proc. ICCV*, 2009, pp. 1420–1427.

[23] A. F. Smeaton, P. Over, and W. Kraaij, "High-level feature detection from video in TRECVid: A 5-year retrospective of achievements," in *Multimedia Content Analysis, Theory and Applications*, A. Divakaran, Ed. Berlin, Germany: Springer-Verlag, 2009.

[24] A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation campaigns and trecvid," in *Proc. MIR*, 2006, pp. 321–330.

[25] A. Natsev, S. Bao, J. Chang, M. Hill, M. Merler, J. R. Smith, D. Wang, L. Xie, R. Yan, and Y. Zhang, "IBM research TRECVID-2009 video retrieval system," in *Proc. TRECVID Workshop*, 2009.

[26] C. G. Snoek, K. E. van de Sande, O. de Rooij, B. Huurnink, E. Gavves, D. Odijk, M. de Rijke, T. Gevers, M. Worring, D. C. Koelma, and A. W. Smeulders, "The MediaMill TRECVID 2010 semantic video search engine," in *Proc. TRECVID Workshop*, 2010.

[27] C. Sun, J. Li, B. Zhang, and Q. Zhang, "THU-IMG at TRECVID 2010," in *Proc. TRECVID Workshop*, 2010.

[28] A. Natsev, S. Bao, J. Chang, M. Hill, M. Merler, J. R. Smith, D. Wang, L. Xie, R. Yan, and Y. Zhang, "Columbia university/vireo-cityu/irit TRECVID2008 high-level feature extraction and interactive video search," in *Proc. TRECVID Workshop*, 2008.

[29] H. Li, L. Bao, Z. Gao, A. Overwijk, W. Liu, L. fei Zhang, S.-I. Yu, M. yu Chen, F. Metze, and A. Hauptmann, "Informedia at TRECVID 2010," in *Proc. TRECVID Workshop*, 2010.

[30] C. G. M. Snoek, M. Worring, J. C. van Gemert, J.-M. Geusebroek, and A. W. M. Smeulders, "The challenge problem for automated detection of 101 semantic concepts in multimedia," in *Proc. ACM Multimedia*, 2006, pp. 421–430.

[31] A. Yanagawa, S.-F. Chang, L. S. Kennedy, and W. Hsu, "Columbia University's Baseline Detectors for 374 LSCOM Semantic Visual Concepts," Columbia Univ., Tech. Rep. 222-2006-8, 2007.

[32] J. Wu and M. Worring, "Genre-specific semantic video indexing," in *Proc. CIVR*, 2010, pp. 266–273.

[33] M. Persin, "Document filtering for fast ranking," in *ACM-SIGIR Special Issue on Research and Development in Information Retrieval)*. New York: ACM/Springer, 1994, pp. 339–348.

[34] D. Carmel, D. Cohen, R. Fagin, E. Farchi, M. Herscovici, Y. S. Maarek, and A. Soffer, "Static index pruning for information retrieval systems," in *ACM SIGIR Special Issue on Research and Development in Information Retrieval*. New York: ACM, 2001, pp. 43–50.

[35] J. Cao, Y. Zhang, Y. Song, Z. Chen, X. Zhang, and J. Li, "MCG-WEBV: A benchmark dataset for web video analysis," Chinese Inst. Computing Technol., Tech. Rep. ICT-MCG-09-001, 2009.

[36] J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih, "Image indexing using color correlograms," in *Proc. CVPR*, 1997, pp. 762–768.

[37] H. Yu, M. Li, H.-J. Zhang, and J. Feng, "Color texture moments for content-based image retrieval," in *Proc. ICIP*, 2003, pp. 24–28.

[38] B. Wang, Z. Li, M. Li, and W.-Y. Ma, "Large-scale duplicate detection for web image search," in *Proc. ICME*, 2006, pp. 353–356.

[39] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool, "A comparison of affine region detectors," *Int. J. Comput. Vision*, vol. 65, no. 1–2, pp. 43–72, 2005.

[40] K. E. van de Sande, T. Gevers, and C. G. Snoek, "Evaluating color descriptors for object and scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1582–1596, Sep. 2010.

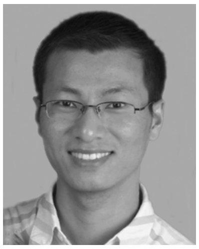
[41] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[42] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Proc. ICCV*, 2003, pp. 1470–1477.

[43] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study," *Int. J. Comput. Vis.*, vol. 73, no. 2, pp. 213–238, 2007.

[44] Y.-G. Jiang, C.-W. Ngo, and J. Yang, "Towards optimal bag-of-features for object categorization and semantic video retrieval," in *Proc. CIVR*, 2007, pp. 494–501.

[45] J. Uijlings, A. Smeulders, and R. Scha, "Real-time visual concept classification," *IEEE Trans. Multimedia*, vol. 12, no. 7, pp. 665–681, Nov. 2010.



Jun Wu received the B.S. degree in information engineering from Xi'an Jiaotong University, Xian, China, in 2001, and the M.Sc. and Ph.D. degrees in computer science and technology from Tsinghua University, Beijing, China, in 2004 and in 2008, respectively.

From 2008 to 2010, he was a Member of Research Staff with the Intelligent Systems Lab Amsterdam, University of Amsterdam, Amsterdam, The Netherlands. He then joined the School of Electronics and Information, Northwestern Polytechnical University, as an Associate Professor. During 2003 to 2004, he was a Visiting Student with Microsoft Research Asia. From August to October in 2005, he was a Visiting Scholar with the Department of Computer Science, University of Hamburg, Hamburg, Germany. His research interests are in machine learning, multimedia analysis, and multimedia information retrieval.

Marcel Worring (M'04) received the M.Sc. degree (with honors) from Vrije Universiteit, Amsterdam, The Netherlands, in 1988, and the Ph.D. degree from the University of Amsterdam, Amsterdam, The Netherlands, in 1993, both in computer science.

He is currently an Associate Professor with the Intelligent Systems Lab Amsterdam, University of Amsterdam, Amsterdam, The Netherlands. His research interests are in multimedia analytics, bringing together multimedia analysis and information visualization. The methodologies he developed are applied to visual search in broadcast archives as well as in the field of forensic intelligence. He has authored and coauthored over 100 peer-reviewed scientific papers covering a broad range of topics from low-level image and video analysis, to multimedia search and analytics. Currently, he is an associate editor of *Pattern Analysis and Applications*.

Prof. Worring was the chair of the IAPR TC12 on Multimedia and Visual Information Systems. He was an associate editor of the IEEE TRANSACTIONS ON MULTIMEDIA. He was co-organizer of the ACM CIVR 2007, and the ACM workshops Multimedia in Forensics, Security and Intelligence 2009, 2010 and 2011. He is a program chair for the ACM Multimedia 2013.