Genre-specific Semantic Video Indexing

Jun Wu, Marcel Worring
Intelligent Systems Lab Amsterdam (ISLA)
Informatics Institute, University of Amsterdam
Science Park 107, 1098 XG Amsterdam, The Netherlands
j.wu@uva.nl, m.worring@uva.nl

ABSTRACT

In many applications, we find large video collections from different genres where the user is often only interested in one or two specific video genres. So, when users are querying the system with a specific semantic concept, they are likely aiming a genre specific instantiation of this concept. Thus, a question is how to detect genre specific semantic concepts such as *Child* in *HomeVideo*, or *FrontalFace* in *Porn*, in an efficient and accurate way. We propose a framework to do such genre-specific context detection. Genre specific models are trained based on a training set with data labelled 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, but realistic 28-hour video data set including YouTube videos, porn videos, TV programs, as well as home videos. Experimental results show that our proposed two-step method is efficient and effective. When filtering the data set such that approximately a percentage is kept equal to the prior probability of each video genre, the overall performance only decreases about 12%, while the processing speed increases about 2 to 10 times for different video genres.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.2.4 [**Database Management**]: Systems—*Multimedia Database*

General Terms

Algorithms, Design, Experimentation

Keywords

Semantic indexing, Genre classification, Efficiency

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
CIVR '10, July 5-7, Xi'an China
Copyright ©2010 ACM 978-1-4503-0117-6/10/07 ...$10.00.

1. INTRODUCTION

Semantic indices to videos can be computed at two different levels, namely at the genre level and the semantic concept level. At the genre level, videos are roughly classified into a couple of pre-existing genres, while at the semantic concept level, video shots are classified by a measure indicating the presence of a given concept. Genres and semantic indices have an intimate relation, some concepts are specific to one genre, where other concept have different visual characteristics in each genre. See Figure 1 for some examples.

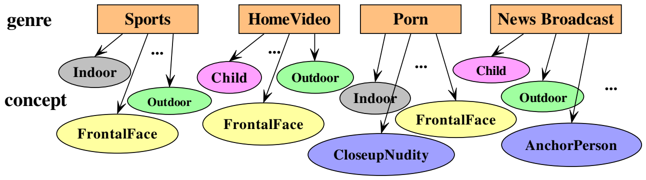


Figure 1: Our goal is to perform genre-specific semantic video indexing, such as detecting *Child* in *HomeVideo*, or *FrontalFace* in Porn.

A video genre is a set of videos sharing similar style [22], which is chosen by the director of the program, and where the style fits the purpose of the genre. For example, popular video genres are news broadcast, home-video, advertisement, music, sports, and movies. In many applications, we find large video collections from different genres where the user is often only interested in one or two specific genres. Our use case is a police investigator who is examining a hard drive of a computer for illegal material, such as *Child Abuse*. Having videos classified into pre-existing genres is one way to make the browsing task easier as it provides the means to browse large scale video sets more efficiently. At the same time it provides a context to further analyze semantic contents of multimedia data.

At the semantic level, research in concept-based video indexing focuses on building large numbers of unrelated individual semantic concept detectors [20, 25, 26] such as *sunsets*, *indoor*, *outdoor*, *cityscape*, *landscape*, *mountains*, and *forests*, or creating a set of concept detectors based on knowledge such as the concept ontology described in [4, 12, 14].

When considering the relation between genres and concepts we should distinguish the following two cases. The first case occurs when a concept is specific to one genre, i.e., it never occurs in other genres. For example, the concept *CloseupNudity* appears only in the *Porn* genre, and the concept *AnchorPerson* shows up only in *News Broadcast*, as illustrated in Figure 1. For such concepts, it is inefficient to train a generic classifier in a traditional way, i.e., using the entire training data. As these concepts only appear in certain genres, we can focus on a genre-relevant subset. So, 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 Figure 2.

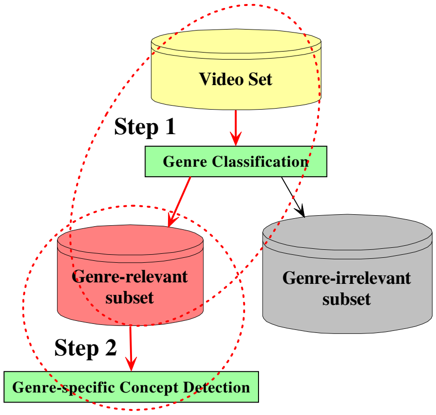


Figure 2: Our two-step framework for genre-specific video indexing. First, we reduce the entire set into a much smaller subset by performing video genre classification to filter out most of the irrelevant videos. Then, the genre-specific concept models are used to classify the estimated subset of the test set, based on the outputs of the video genre classification.

In the second case, one concept might occur in several genres. For example, *Indoor* or *Outdoor* can be in *Broadcast*, *HomeVideo*, *Sports*, and *Porn*. For such concepts, there are often large variations among different genres, resulting in diverse visual appearances. For instance, in a movie people never look into the camera directly, whereas in home video or in mobile phone video talking heads are appearing quite often. We call such concepts *genre-specific concepts*. Obviously, this variation becomes less if we restrict the analysis to videos in a particular genre type hence concept detection becomes easier. For example, the *Table* in the *Meeting* genre is more restricted than a general table. When the users only care about a certain concept within a target genre, the generic concept models are prone to be under-fitting the data. To improve, genre-specific concept models need to be derived from the subset within the specific genre.

As explained above, 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, as illustrated in Figure 2. In the first first stage, genre-specific concept models are trained based on the data from the target video genre. In the classification stage, we first perform video genre classification to filter out most of the irrelevant videos. The genre-specific models are then used to classify the remaining videos. Of course this introduces the risk of throwing away too much videos so we need to carefully handle the trade-off between accuracy and efficiency.

The rest of this paper is organized as follows. In Section 2 we review related work. We introduce our framework in Section 3. In Section 4, extensive experiments are presented, followed by conclusions and future work in Section 5.

2. RELATED WORK

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

2.1 Video Genre Classification

A *video genre* is a set of video documents sharing similar style [22], such as broadcast, home-video, or movie. Work on automatic video genre categorization begins with Fisher [5] in 1995. Till now, most of the existing work focuses on movie, TV [16, 31, 34] or online videos [2, 24, 30, 33].

Yuan [34] presents an automatic video categorization scheme based on a *hierarchical ontology* of video genres, in which the basic genres are: movie, commercial, news, music video, sports, and so on; then movie and sports are further divided into a couple of sub-classes. A *Hierarchical* SVM united in a binary-tree form is effective for the generic problem of video genre classification, but ignores the fact that users might be interested in a limited number of genres for further analysis. The complexity of the hierarchical genre structure might yield low performance for a set of limited video genres.

To enable taxonomy-based browsing, Borth [2] presents a framework called TubeFiler to categorize web videos based on a 46-category genre hierarchy. The first-level seven categories are classified based on metadata such as tags and titles. Then, the second-level 39 sub-categories are fine-grained automatically based on visual features. This method performs well when sufficient metadata is present, but in cases where this is lacking, like our use case, a content-based solution is required.

2.2 Semantic Indexing

Extracting the semantics of videos is one of the most important tasks in *content-based semantic indexing*. Till now, many concept detectors have been obtained using different pattern recognition techniques. In early literature [20, 25, 26], specific concept detectors, such as *news anchor person*, *sunsets*, *indoor*, *outdoor*, *mountains*, and *forests*, have been developed specifically developed for the target class. Other work explores the relationships between semantic concepts, such as hierarchical models [4], the co-occurrence of two concepts [13], actions or objects in context [10, 15], and inter-concept relationships [8, 14, 29].

The TREC Video Retrieval Evaluation (TRECVID) conference [18, 19], starting from 2001, started providing a large test collection as a benchmark for all participants. Among others, systems such as IBM system [1], MediaMill [21], the Video Diver System from Tsinghua [28], the Columbia system [3], and the Informedia system from CMU [6], have been developed for this task. With more and more powerful computing resources, detecting a large amount of concept detectors is now feasible [12, 23, 32]. The TRECVID has started the trend to move from specific purpose build models, to generic models suited for every genre. Thus, methods require complex statistical models to cover the large variations in appearance over the different genres. We feel there is a need to have models which are trained in a generic way, but which are genre specific, hence they can in many cases be based on simpler models.

Our goal is different from the traditional concept detection as introduced above, which considers the concepts in broader domains, i.e., ignoring the information of the video genres. We mainly consider the efficiency for detecting genre-specific concept, using relations such as video genres and semantic concepts at different semantic level, as illustrated in Figure 1. When the users only care about a certain concept within a target video genre, the generic concept models are prone to be under-fitting in such a narrow domain, i.e., within the target genre. Consequently, the genre-specific concept models need to be derived from the data subset within the specific video genre, which are quite different from the models using traditional training strategies.

3. THE PROPOSED FRAMEWORK

We now detail the framework to predict genre-specific concepts. Suppose there are in total J target 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

$$C_G = \{c_{1,g_1}, \cdots, c_{k,g_j}, \cdots, c_{K,g_J}\} ,$$

with c_{k,g_j} a genre-specific concept defined as the concept $c_k \in C$ within the video genre $g_j \in G$.

Further, assume we have a video set $\mathbf{V}$, divided into two subsets: a training set $\mathbf{V_X} = \{V_1, V_{2,}, \cdots, V_{M_X}\}$ and a test set $\mathbf{V_Y} = \{V'_1, V'_{2,}, \cdots, V'_{M_Y}\}$. Shot segmentation is conducted per video on both training and test set, then key-frames are extracted within each shot, resulting in two data sets, $\mathbf{X} = \{X_1, X_2, \cdots, X_{N_X}\}$ for the training set V_X and $\mathbf{Y} = \{Y_1, Y_2, \cdots, Y_{N_Y}\}$ for the test set V_Y, respectively.

As ground truth let $Q_g : \mathbf{X} \cup \mathbf{Y} \rightarrow G$ be a mapping from a given sample X_i or Y_i to its real genre, e.g g_j. Similarly, let $Q_c : \mathbf{X} \cup \mathbf{Y} \rightarrow C$ be a mapping from a given sample X_i or Y_i to its real concept, e.g c_k. For the J video genres, video-based annotations are executed, i.e., each video is assigned a positive or negative label. All the shots within a single video share the same genre type. Thus, for the target genre g_j and for each genre-specific concept c_{k,g_j}, all the key-frames in each shot are labelled as positive or negative samples.

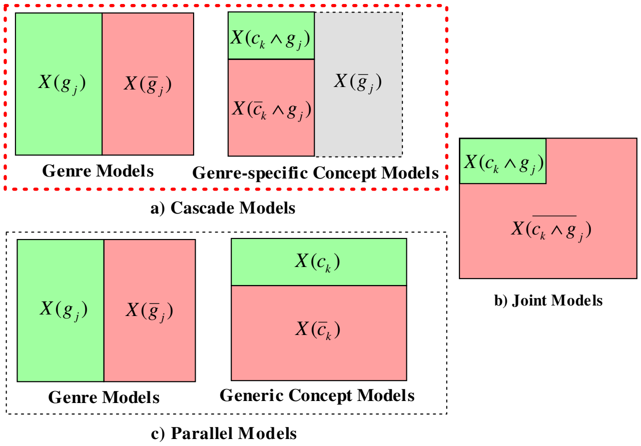
For a given target, which can be a genre g_j or a genre-specific concept c_{k,g_j}, we build a model $M(\cdot)$ based on a set of annotated examples $X(\cdot)$ which are a subset of $\mathbf{X}$ defined by the argument of M. With the model we can compute the probability $P(c_k|Y_i)$ given a sample Y_i in the test set $\mathbf{Y}$. Our goal is to predict the posterior probability of the genre-specific concept $P(c_{k,g_j}|Y_i)$. Note, as several concepts are restricted to one specific genre, many of these will have zero probability of occurrence.

In the following we will step by step introduce the training stage and classification stage, as illustrated in Figure 4–5.

3.1 Training Stage

To predict a given genre-specific concept c_{k,g_j}, we use a two-step strategy: video genre classification is first applied to filter out most of the irrelevant videos, then the genre-specific concept models classify the samples within the estimated subset for the given video genre. We call these *cascade models*. The genre-specific concept models $M(c_{k,g_j})$ within the target genre g_j can be trained from the annotations as in Figure 3-a (cascade models). Note $X(c_k \wedge g_j)$ denotes the positive sample set, and $X(\overline{c_k} \wedge g_j)$ means the negative sample set for the concept model $M(c_{k,g_j})$.

To compare the cascade models to other possible options, we also train the genre-and-concept *joint model* $M(c_k \wedge g_j)$ directly from the data annotations as in Figure 3-b. A shot for both positive genre and positive concept is regarded as a positive genre-and-concept shot; otherwise, it is considered as a negative shot. Note that $X(c_k \wedge g_j)$ is the positive sample set, while $X(\overline{c_k \wedge g_j})$ denotes the negative sample set. Another choice is to combine a genre model and a generic concept model, e.g. by applying rank fusion. As shown in Figure 3-c, these two models can be trained in parallel, using the training set $X(g_j) \cup X(\overline{g_j})$ for the genre model and $X(c_k) \cup X(\overline{c_k})$ for the generic concept model, respectively. So, we call both of them *parallel models*. Figure 4 illustrates the above three categories of models in the training stage.



a) Cascade Models

b) Joint Models

c) Parallel Models

Figure 3: Three schemes to train genre-specific concept models: a) cascade models, b) joint models, and c) parallel models. In each sub-figure, the whole rectangle denotes all possible shots and their distribution (green for positive samples, red for negative samples, and gray for ignored samples) for a genre g_j, a concept c_k, or a genre-specific concept c_{k,g_j}.

3.1.1 Cascade Models

Following the above ideas, as illustrated in Figure 3-a, based on the annotations of the J video genres, a set of genre models can be trained as

$$\mathbf{M}_G = \{M(g_1), M(g_2), \cdots, M(g_j), \cdots, M_{g_J}\} , \quad (1)$$

where $M(g_j) = P(g_j|X)$. For each genre g_j, we train a set of genre-specific concept models within all the positive videos for the genre g_j, i.e., all the negative videos will be ignored. The genre-specific concept models are

$$\mathrm{M}^S(g_j) = \{M^S(c_{1,g_j}), M^S(c_{2,g_j}), \cdots, M^S(c_{k,g_j})\} , \quad (2)$$

where $M^S(c_{k,g_j}) = P(c_k|X_{g_j})$. Note that the subset within the genre g_j is $X_{g_j} = \{X_i : Q_g(X_i) = g_j\}$. Accordingly, the full set of genre-specific concept models is given by

$$\mathbf{M}^S(G) = \{\mathrm{M}^S(g_1), \mathrm{M}^S(g_2), \cdots, \mathrm{M}^S(g_J)\} . \quad (3)$$

3.1.2 Joint Models

For each video genre g_j, based on the annotations for a genre-and-concept pair $c_k \wedge g_j$, we train a set of joint models for the given genre g_j, as

$$\mathrm{M}^J(g_j) = \{M^J(c_1 \wedge g_j), M^J(c_2 \wedge g_j), \cdots, M^J(c_k \wedge g_j)\} , \quad (4)$$

where $M^J(c_k \wedge g_j) = P(c_k \wedge g_j|X)$. Accordingly, genre-and-

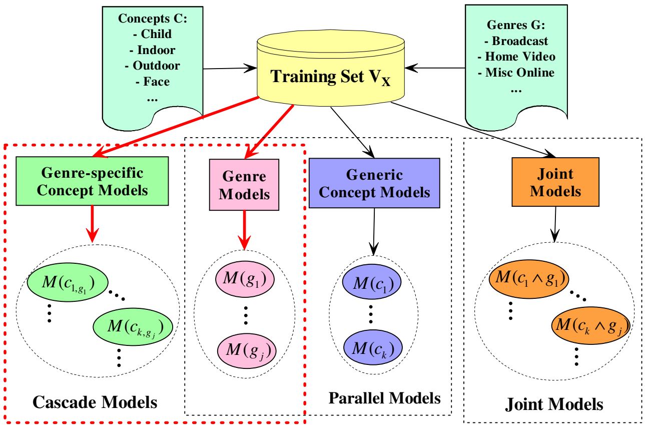


Figure 4: A flowchart of the training stage. In the cascade models, the genre-specific concept models $M(c_{k,g_j})$ (left) are trained only using samples within the target genre g_j; in the parallel models (middle), the genre models $M(g_j)$ and generic concept models $M(c_k)$ can be trained in parallel; the joint models $M(c_k \wedge g_j)$ (right) are trained based on the annotations for the genre-and-concept pairs $c_k \wedge g_j$.

concept joint models are given by:

$$\mathbf{M}^J(G) = \{\mathrm{M}^J(g_1), \mathrm{M}^J(g_2), \cdots, \mathrm{M}^J(g_J)\}\ . \qquad (5)$$

3.1.3 Parallel Models

Based on the annotations of the K semantic concepts, we train a set of generic concept models

$$\mathbf{M}_C = \{M(c_1), M(c_2), \cdots, M(c_k), \cdots, M(c_K)\}\ , \qquad (6)$$

where $M(c_k) = P(c_k|X)$. In addition, for the parallel models, the genre models are the same as defined in equation (1) for the cascade models.

3.2 Classification Stage

We use a two-step strategy (cascade models) to predict genre-specific concepts: video genre models are first applied to filter out most of the irrelevant videos, then the genre-specific concept models classify the samples within the estimated subset for the given video genre. We also compare cascade models to parallel models and joint models.

We apply all the available models from the previous training stage on the test set $\mathbf{Y}$, resulting in series of posterior probabilities $P(\cdot)$ given the i-th test sample in the set $\mathbf{Y}$, which are listed in Table 1. Based on these posterior probabilities, we can achieve the final scores for these three groups of models, as shown in Figure 5.

Table 1: The posterior probabilities of different models, in three categories: 1) cascade models, 2) joint models, 3) parallel modes.

Name	Model	Post-Prob.	Category
genre-specific	$M^S(c_{k,g_j})$	$P(c_{k,g_j}\|Y_i)$	1)
joint model	$M^J(c_k \wedge g_j)$	$P(c_k \wedge g_j\|Y_i)$	2)
genre model	$M(g_j)$	$P(g_j\|Y_i)$	1) or 3)
concept model	$M(c_k)$	$P(c_k\|Y_i)$	3

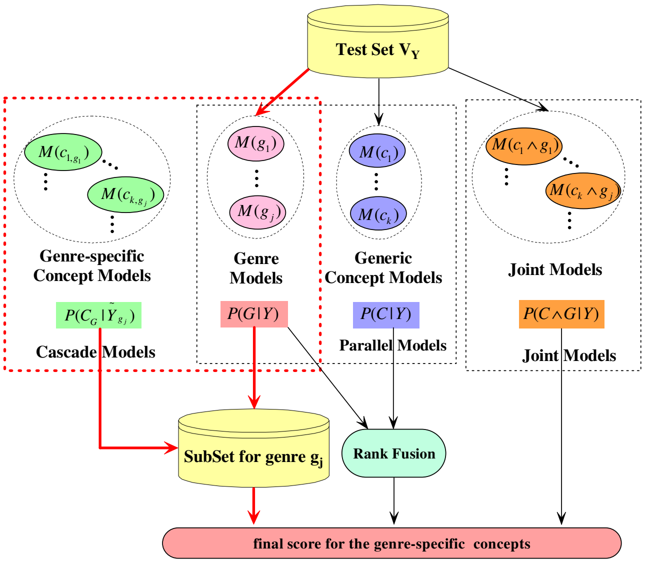


Figure 5: A flowchart of the classification stage. The genre models (middle) are first applied to the entire test set, then the genre-specific concept models (left) classify shots within the estimated subset. Rank fusion is conducted combining the outputs of the genre models and joint models. The joint models (right) are employed in the entire test set.

3.2.1 Applying Cascade Models

To enable speed-up in detecting genre-specific concepts, the video genre models are applied to achieve a subset of the test set for the target genre. More precisely, the genre model $M(g_j)$ results in shot-based scores $P(g_j|Y_i)$. For the target genre g_j, we discard *irrelevant* video shots, yielding a relatively-small subset of the test set

$$\tilde{Y}_{g_j} == \{Y_i : P(g_j|Y_i) > \beta\}\ , \qquad (7)$$

where the parameter β is a threshold which controls the size of remaining subset. For example, a certain percentage can be kept according to the prior probability of each video genre. Based on this estimated subset $\tilde{Y}_{g_j}$, we apply the genre-specific model $M^S(c_{k,g_j})$ to obtain the posterior probability of the genre-specific concept, $P_{M^S}(c_{k,g_j}|\tilde{Y}_{g_j})$.

3.2.2 Applying Parallel Models and Joint Models

A straightforward way to combine a genre model and a concept model into a genre-specific model is using rank fusion. We combine the posterior probability of the genre g_j, $P(g_j|\tilde{Y}_{g_j})$, and that of the concept c_k, $P(c_k|\tilde{Y}_{g_j})$, using simple multiplicative fusion

$$P(c_{k,g_j}|Y) = P(g_j|Y) * P(c_k|Y)\ . \qquad (8)$$

In addition, we apply the joint model $M^J(c_k \wedge g_j)$ to retrieve the posterior probability of the genre-and-concept pair

$$P_{M^J}(c_k \wedge g_j|Y)\ . \qquad (9)$$

3.3 Discussion

For real-world applications, an efficient way to detect the genre-specific concepts is to use cascade models. For example, if we detect the concept *CloseupNudity*, there is no need to apply its corresponding classifier on the entire video set in a traditional way. We only need to focus on the videos belonging to the *Porn* genre, because the concept *Closeup-Nudity* only happens in *Porn* videos.

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. For example, in our development set, there are more than 7,200 key-frames but only about 250 videos, resulting in at least 20 times' speed-up. Further, the video genre classification can be handled in a more efficient way. It is possible to use cheap features for doing genre classification and then compute the more complex and expensive features for a subset only. In addition, the shot-based processing can be replaced by randomly or uniformly sampled frames from each video, skipping shot segmentation.

4. EXPERIMENTS

We now experimentally verify the efficiency and effectiveness of our two-step framework in detecting genre-specific concepts. We first show the performance of video genre classification, then we evaluate the genre-specific models. For the efficiency, we consider how much time can be saved when applying the two-step cascade models in Section 3.1.1, and at what loss in performance. For evaluating the effectiveness, we compare the cascade models with the joint models from Section 3.1.2 and parallel models from Section 3.1.3.

4.1 Data Set and Basic Setups

As indicated, our use case is a police investigator who is examining a hard drive of a computer for illegal materials, such as videos containing *Child Abuse*. To implement this task, we use a small, but realistic 28-hour data set (474 videos in total), including some TV programs, home videos, porn videos, as well as some online videos. This video set is split into two subsets: 245 videos (15 hours, about 6 G) for training, and another 229 videos (13 hours, about 5 G) for testing. The size of videos exhibit large variations: from 2 seconds to 31 minutes. In this data set, 382 videos (81%) are shorter than 5 minutes, and only 41 videos (9%) are larger than 8 minutes.

Each videos is first segmented into video shots, and representative key-frames are selected from each shot. Further, visual features are extracted based on the representative key-frames. The dense sampling detector [11] and Harris-Laplace salient point detector are applied both in the opponent color space. Then, OpponentSIFT [27] features, a variant of SIFT [9], are extracted based on a spatial pyramid with a 1×1, 2×2 and 1×3 layout. The Bag of Words model [17] is employed based on a visual vocabulary of 3961 visual words for the dense sampling detector, and a visual vocabulary of 3950 visual words for the Harris-Laplace detector. Both of the codebooks are constructed based on the TRECVID [19] 2007 development set. The Support Vector Machine with a χ^2 kernel [7, 35] is used for learning. In addition, parameter tuning for the SVM is conducted on the training set using a 3-fold cross validation. In addition, the tuning of the parameter β in equation (7) can be conducted on the training set beforehand.

4.2 Genre-specific Concept Annotation

Six video genres ($g_1 \sim g_6$: *Broadcast*, *HomeVideo*, *Porn*, *MiscOnline*, *Sports*, and *Soap*) are manually labelled at video level. Note that *MiscOnline* means the whole data set excluding the other five video genres. This rest set mainly includes some small videos downloaded from the Internet. 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 our use case, the investigators are interested in specific concepts such as *Closeup-Nudity* 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.

Since some concepts are restricted to one specific genre, we illustrate the conditional probability $P(c_k|g_j)$ of the target concept c_k given the video genre g_j in Figure 6. We ignore all the genre-and-concept pairs with too few positive samples, set in this case as less than 50, because too few positive sample in the training set makes learning genre-specific concept models inaccurate. As a result, 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*.

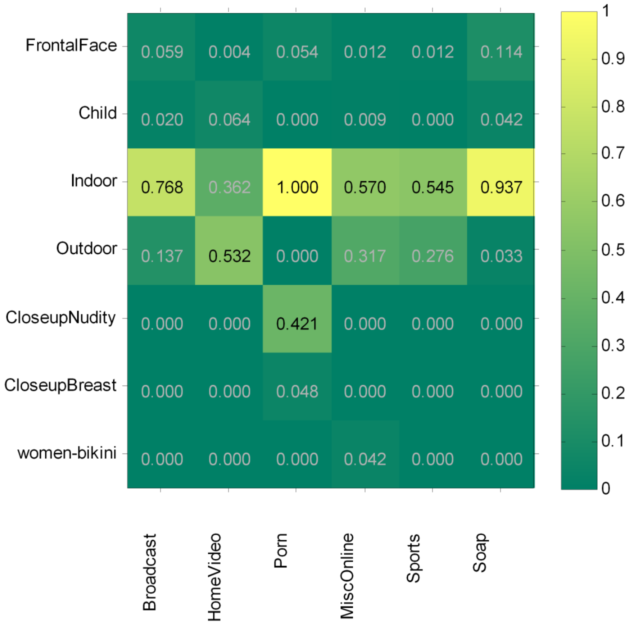


Figure 6: The conditional probabilities $P(c_k|g_j)$ for the concept c_k (y-axis) given the genre g_j (x-axis).

4.3 Results of Genre Classification

As described above, video genre classification is the first step of the 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 full list of video shots (AP) and in the top 2000 returned video shots (AP2000) respectively, as shown in Table 2. The Mean Average Precision (MAP) for the current test set is about 0.838. Based on the current performance, we can conclude that the

Table 2: The Average Precision of Video Genre Classification. AP denotes the Average Precision evaluated in the full list, while AP2000 means the Average Precision evaluated in the top 2000 shots.

g_j	Genre	Prior	AP	AP2000
g_1	Broadcast	25.7%	0.728	0.580
g_2	HomeVideo	13.9%	0.981	0.979
g_3	Porn	11.7%	0.991	0.990
g_4	MiscOnline	48.8%	0.905	0.568
g_5	Sports	9.4%	0.688	0.668
g_6	Soap	16.3%	0.734	0.670

step of video genre classification can provide a good starting point for genre-specific concept detection.

4.4 Cascade Model Evaluation

As mentioned above, the two-step cascade models save large amount of human labor for annotating training data. They also speed up the investigation 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, without losing too much in performance.

The efficiency curve of the cascade models, i.e., MAP vs. how large a percentage is kept from the test data set is illustrated in Figure 7. When keeping the full test set (100%), the highest MAP which can be achieved is 0.478. When using the two-step cascade models to filter-out 50% of the data, the MAP decreases about 3% (0.464). When 20% of the data are kept for further processing, i.e., 80% of the data are ignored, the MAP decreases about 11% (0.424). Even when 90% of the data are ignored, the MAP decreases less than 30% (0.346).

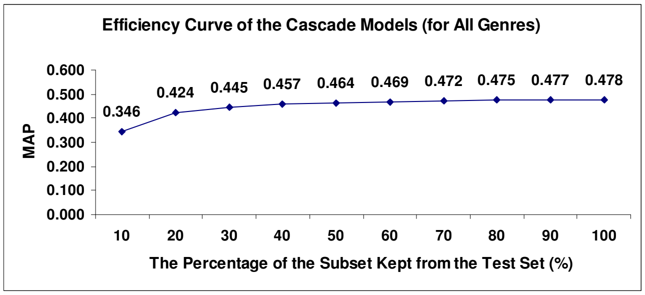


Figure 7: The efficiency curve of the cascade models. The x-axis denotes the percentage of the test data set kept, while the y-axis denotes Mean Average Precision (MAP). We consider 15 genre-specific concepts in all the six video genres.

Figure 7 shows the results for all the six genres together. However, the prior probabilities for different video genres vary, as listed in Table 2. So, we show similar efficiency curves for each video genre separately in Figure 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 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.

4.5 Comparison with Other Schemes

Next to the cascade models, we also implemented the other two methods defined in Section 3.1.2 and 3.1.3. We compare these three kinds of models. Briefly speaking, the cascade models, namely scheme A1), train the genre-specific concept models within the target genre, and classify the subset belonging to the target genre. The joint models, namely scheme B, train the genre-and-concept joint models based on the annotations for the genre-and-concept pairs. The parallel models, namely scheme C, fuse the results from the video genre classification and the generic concept detection. 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 a variant of cascade models as scheme A2, which is using the ground truth information for each video genre.

The performance of these different schemes is listed in Figure 9. Since the genre-specific concept models only classify a subset of the test set, we consider the Mean Average Precision (MAP) in the top 500 shots (AP500) and top 1000 shots (AP1000), respectively. From this figure, 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. Further, the cascade models (A1) are consistently better than the parallel models (C) and the joint models (B). We conclude that our two-step cascade models are effective.

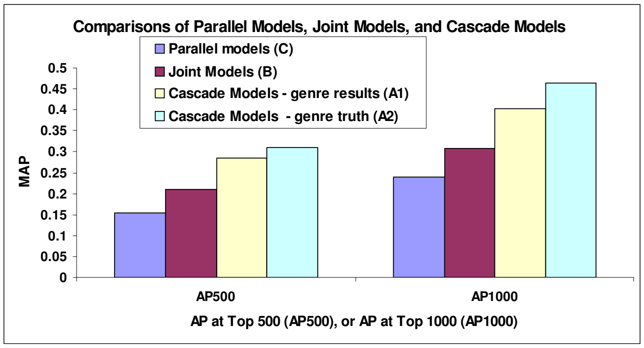


Figure 9: Comparing the cascade models to the parallel models, the joint models, and a variant of cascade models using the genres' ground truth (instead of using the predicted genre results). The Mean Average Precision in the first 500 (left group) and 1000 (right group) shots are listed.

4.6 Discussion

As introduced in Section 1, there are two different types of genre-specific concepts. In the first case, a concept occurs only within a specific video genre, such as *Porn-CloseupNudity*. In the second case, a concept might occur in any genres, such as the *Indoor* and *Outdoor* in different genres. As shown in Figure 7-c), the efficiency improvement of the first case is significant. For the second case, *HomeVideo* and *soap* perform the best. Though the efficiency improvement of *MiscOnline* is not so good as for the other video genres, its performance

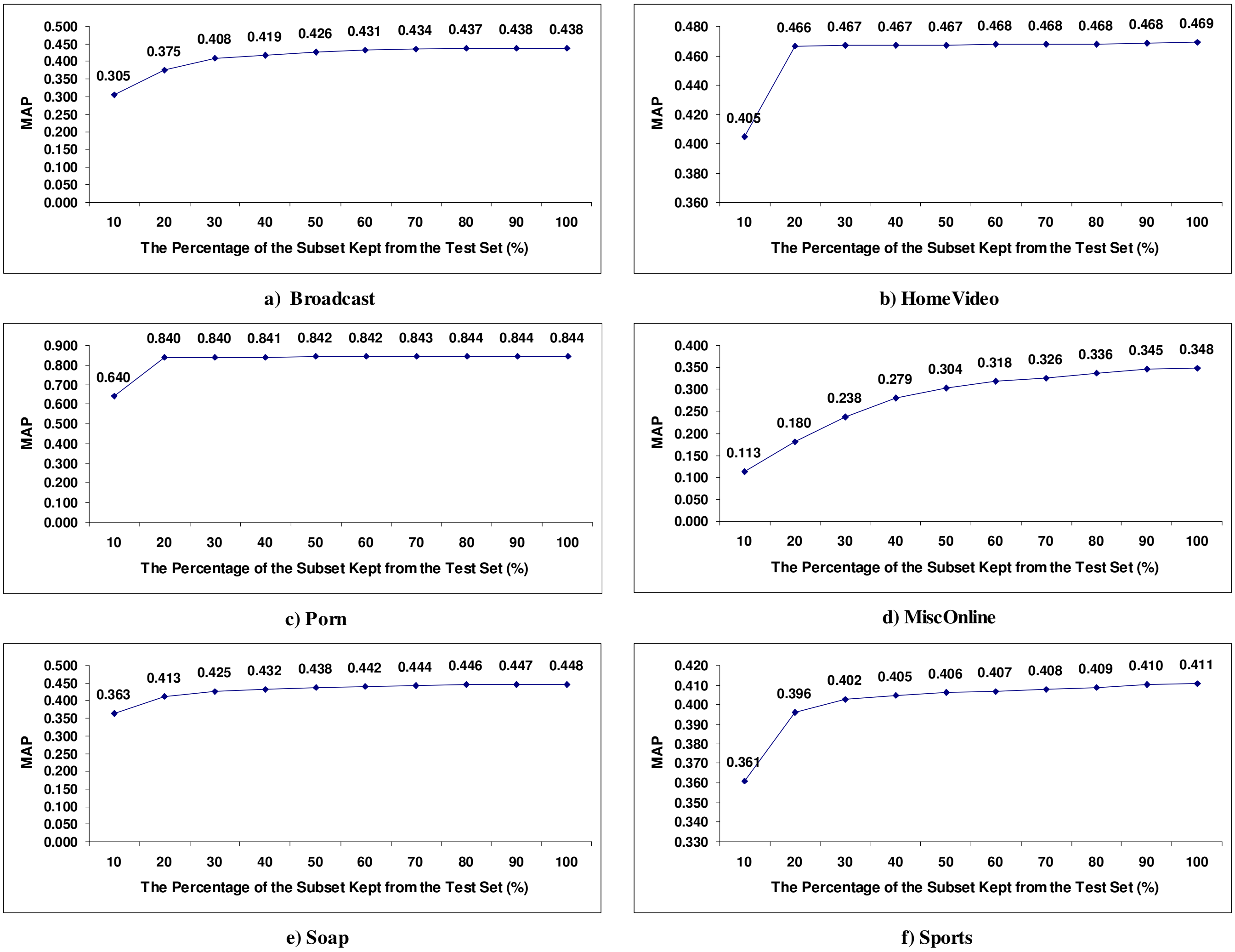


Figure 8: The efficiency curves of the cascade models. The x-axis denotes which percentage of the test data set is kept; the y-axis denotes Mean Average Precision (MAP). We consider each video genre separately, resulting in six sub-figures. These sub-figures show the results for six video genres from a) to f).

is still reasonable. One reason is that the prior probability of *MiscOnline*, almost 50%, is higher than other genres in the test set. So, its intra-genre diversity is still large. A possible solution is to further split *MiscOnline* into several sub-genres to make the intra-genre variation smaller.

5. CONCLUSION

In this paper we propose a two-step framework to detect genre-specific concepts, such as *Child* in *HomeVideo*, or*FrontalFace* in *Porn*. 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 models, which are trained within each given video genre, classify the videos in the estimated subset only. Experimental results show that our two-step method is efficient and effective. If ignoring a percentage of different video genres equal to their priors, the overall performance only decreases about 12%, while the processing speed is between 2 to 10 times faster depending on the video genre. If keeping the corresponding percentage of the different video genres at twice their prior probabilities, the overall performance only decreases about 2%. This two-step method also performs better than the fusion of the parallel models (the genre models and generic concept models) and the genre-and-concept joint models. As a result, we conclude that our framework provides an efficient way to detect genre-specific concepts.

In future work, we will test our framework on a large-scale video data set. Applying the two-step cascade models, it is possible to use cheap features for doing genre classification and then compute the more complex and expensive features for a subset only. For example, the shot-based processing can be replaced by randomly or uniformly sampled frames from each video, skipping shot segmentation for the step of video genre classification. This will further speed up the process of finding the genre-specific concepts.

6. ACKNOWLEDGMENTS

This research was co-funded by the EU project: The Investigator's Dashboard, through the Safer Internet plus program, SIP-2007-TP-1317033.

7. REFERENCES

[1] A. Amir, M. Berg, S.-F. Chang, W. Hsu, G. Iyengar, C.-Y. Lin, M. R. Naphade, A. P. Natsev, C. Neti, H. J. Nock, J. R. Smith, B. L. Tseng, Y. Wu, and D. Zhang. IBM research TRECVID-2003 video retrieval system. In *TRECVID Workshop*, 2003.

[2] D. Borth, J. Hees, M. Koch, A. Ulges, C. Schulze, T. Breuel, and R. Paredes. Tubefiler: an automatic web video categorizer. In *ACM Multimedia*, 2009.

[3] S.-F. Chang, W. Jiang, A. Yanagawa, and E. Zavesky. Columbia university TRECVID2007 high-level feature extraction. In *TRECVID Workshop*, 2007.

[4] J. Fan, H. Luo, Y. Gao, and R. Jain. Incorporating concept ontology for hierarchical video classification, annotation, and visualization. *IEEE Transactions on Image Processing*, 9(5):939–957, 2007.

[5] S. Fischer, R. Lienhart, and W. Effelsberg. Automatic recognition of film genres. In *ACM Multimedia*, 1995.

[6] A. G. Hauptmann, M.-Y. Chen, M. G. Christel, C. Huang, W.-H. Lin, T. Ng, N. Papernick, A. Velivelli, J. Yang, R. Yan, H. Yang, and H. D. Wactlar. Confounded expectations: Informedia at TRECVID 2004. In *TRECVID Workshop*, 2004.

[7] Y.-G. Jiang, C.-W. Ngo, and J. Yang. Towards optimal bag-of-features for object categorization and semantic video retrieval. In *CIVR*, 2007.

[8] Y.-G. Jiang, J. Wang, S.-F. Chang1, and C.-W. Ngo. Domain adaptive semantic diffusion for large scale context-based video annotation. In *ICCV*, 2009.

[9] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004.

[10] M. Marszałek, I. Laptev, and C. Schmid. Actions in context. In *CVPR*, 2009.

[11] 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*, 65(1-2):43–72, 2005.

[12] 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*, 13:86–91, 2006.

[13] M. R. Naphade, I. Kozintsev, and T. Huang. Probabilistic semantic video indexing. In *NIPS*, 2000.

[14] A. P. Natsev, A. Haubold, J. Tešić, L. Xie, and R. Yan. Semantic concept-based query expansion and re-ranking for multimedia retrieval. In *ACM Multimedia*, 2007.

[15] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. In *ICCV*, 2007.

[16] Z. Rasheed, Y. Sheikh, and M. Shah. On the use of computable features for film classification. *IEEE Trans. on Circuits and Systems for Video Technology*, 15(1):52–64, 2005.

[17] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, 2003.

[18] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and trecvid. In *MIR*, 2006.

[19] A. F. Smeaton, P. Over, and W. Kraaij. High-Level Feature Detection from Video in TRECVid: a 5-Year Retrospective of Achievements. In A. Divakaran, editor, *Multimedia Content Analysis, Theory and Applications*. Springer Verlag, 2009.

[20] J. R. Smith and S.-F. Chang. Visually searching the web for content. *IEEE MultiMedia*, 4(3):12–20, 1997.

[21] C. G. M. Snoek, I. Everts, J. C. van Gemert, J.-M. Geusebroek, B. Huurnink, D. C. Koelma, M. van Liempt, O. de Rooij, K. E. A. van de Sande, A. W. M. Smeulders, J. R. R. Uijlings, and M. Worring. The MediaMill TRECVID 2007 semantic video search engine. In *TRECVID Workshop*, 2007.

[22] C. G. M. Snoek and M. Worring. Multimodal video indexing: A review of the state-of-the-art. *Multimedia Tools and Applications*, 25(1):5–35, 2005.

[23] 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 *ACM Multimedia*, 2006.

[24] 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 *ACM Multimedia*, 2009.

[25] A. Vailaya, M. A. T. Figueiredo, A. K. Jain, and H.-J. Zhang. Image classification for content-based indexing. *IEEE Transactions on Image Processing*, 10(1):117–130, 2001.

[26] A. Vailaya, A. K. Jain, and H.-J. Zhang. On image classification: City images vs. landscapes. *Pattern Recognition*, 31(12):1921–1936, 1998.

[27] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (in press), 2010.

[28] D. Wang, X. Liu, L. Luo, J. Li, and B. Zhang. Video diver: generic video indexing with diverse features. In *MIR*, 2007.

[29] X.-Y. Wei, Y.-G. Jiang, and C.-W. Ngo. Exploring inter-concept relationship with context space for semantic video indexing. In *CIVR*, 2009.

[30] X. Wu, W.-L. Zhao, and C.-W. Ngo. Towards google challenge: combining contextual and social information for web video categorization. In *ACM Multimedia*, 2009.

[31] L.-Q. Xu and Y. Li. Video classification using spatial-temporal features and PCA. In *ICME*, 2003.

[32] A. Yanagawa, S.-F. Chang, L. S. Kennedy, and W. Hsu. Columbia university's baseline detectors for 374 LSCOM semantic visual concepts. Technical Report 222-2006-8, 2007.

[33] L. Yang, J. Liu, X. Yang, and X.-S. Hua. Multi-modality web video categorization. In *MIR*, 2007.

[34] X. Yuan, W. Lai, T. Mei, X. sheng Hua, X. qing Wu, and S. Li. Automatic video genre categorization using hierarchical SVM. In *ICIP*, 2006.

[35] 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. Vision*, 73(2):213–238, 2007.