(Un)Reliability of Video Concept Detection

Jun Yang
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213, USA
juny@cs.cmu.edu

Alexander G. Hauptmann
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213, USA
alex@cs.cmu.edu

ABSTRACT

Great effort has been made to improve video concept detection and continuous progress has been reported. With the current evaluation method being confined to carefully annotated domains and thus quite forgiving, the reliability of the state-of-the-art concept classifiers remains in question. Adopting a more rigorous evaluation approach, we find that most concept classifiers built using the mainstream approach are unreliable because they generalize poorly to domains other than their training domain. Moreover, evidences show that SVM-based concept classifiers learn little beyond memorizing most of the positive training data, and behave close to memory-based models such as kNN indicated by comparable performance between the two models. Examining the properties of the reliable concept classifiers, we find that the classifiers of frequent concepts, "bloated" classifiers, and classifiers capable of learning the pattern of data, tend to be more reliable. This paper contributes to a better understanding of concept detection, suggests heuristics to identify reliable concept classifiers, and discusses solutions to improving concept detection reliability.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing

General Terms

Experimentation, Performance

Keywords

Video concept detection, Generalizability, SVMs, kNN

1. INTRODUCTION

Semantic concept detection is a challenging research topic critical to the analysis and retrieval of multimedia data. Also known as high-level feature extraction, it aims to determine the presence or absence of any semantic concepts, such as *Outdoor*, *Studio*, and *Airplane*, in images or video shots [11]. The basic approach of concept detection is to use classification algorithms, typically support vector machines (SVMs), to build *concept classifiers* which predict the relevance between images or video shots and a given concept. Built on this basic approach, many sophisticated methods have been proposed in the recent years that exploit, among other things, heavy parameter tuning, concept correlations [7, 10, 13], and combination of different features and models [1, 8, 12, 15]. Continuous performance improvement has been reported using these methods.

What is the real status of the state-of-the-art concept detection methods? An objective evaluation method is key to the answer. In the literature, most concept detection methods are evaluated against a specific TRECVID benchmark dataset which contains broadcast news video or documentary video. A concept classifier is evaluated by *average precision* (AP), which examines whether positive data are ranked higher than negative ones based on the classifier's output, and MAP as the mean of APs. Despite its popularity, this evaluation setting has several limitations:

- First, the training and test data of concept classifiers are typically drawn from a single domain of relatively homogeneous data, such as news video from same broadcaster(s) produced in a certain period of time, and rarely from different domains. This creates room for overfitting and illusions of good performance. Therefore, the previous evaluations can be too forgiving. How well concept classifiers generalize across different domains is a tougher but more faithful metric.

- Second, AP as a metric of classifier performance is affected by the frequency of a concept in the sense that a frequently-occurring concept has a higher "random baseline" than a rare concept. As we will see, this makes results of different concept classifiers less comparable and performance improvement deceptive.

- Last but not the least, most existing work on concept detection is result-driven. This is risky when the evaluation method itself is limited. Little effort has been spent on finding out why a concept classifier works and what it has learned, besides knowing its performance in terms of a certain metric.

Limiting as this evaluation method is, re-examining concept detection methods regarding the above issues is important to understand the real status of this area.

CIVR'08, July 7–9, 2008, Niagara Falls, Ontario, Canada.

This paper provides more rigorous evaluations and deeper examinations of concept classifiers built using the mainstream approach. The main observation is the poor generalization ability of concept classifiers. The majority of the concept classifiers suffer a significant and consistent decline of performance when they are applied to a domain other than their training domain, whether it be another channel (broadcaster) or another genre. Another observation is that SVM concept classifiers in general have learned little from the data besides memorizing the positive instances, because over 90% of the positive instances are support vectors (SVs). They behave close to memory-based models such as k-nearest neighbor (kNN), indicated by the comparable performance between the two models. Together, these observations show that, in general, concept classifiers are not as reliable or intelligent as previous evaluations might have suggested.

While concept classifiers are generally unreliable, some are more reliable than the others. We provide an in-depth analysis as to which concept classifiers are more reliable, and more importantly, the common properties of these reliable classifiers. We find that classifiers of frequent concepts, classifiers bloated with a large number of SVs, and classifiers using a small percentage of positive data as SVs have relatively consistent and generalizable performance. These heuristics help us identify reliable concept classifiers *a priori* without testing them on all possible domains, which is prohibitively expensive in terms of labeling effort.

This paper does not evaluate *any* specific method for concept detection; instead, we adopt the general approach to build concept classifiers using SVMs with RBF kernel and empirically sound model parameters. The state-of-the-art methods, which involve heavy parameter tuning, feature engineering, and/or exploiting of other knowledge [1, 8, 12, 10, 13, 15], almost certainly achieve better performance (in terms of AP or MAP) than the concept classifiers used in this study. Nevertheless, our findings should still be valid because our approach is the very core and the building block of more sophisticated methods. In fact, more tweaking of the concept classifiers may result in overfitting and worse generalization ability despite achieving higher APs or MAPs on a specific data set.

Rather than proposing a new method, our paper contributes to a better understanding of concept detection. First of all, it presents a quantitative study on the performance of concept classifiers in cross-domain settings, which shows how unreliable and fragile they are. While this issue has been discussed in some previous work (e.g., [4]), our study is more comprehensive. We are also the first to examine the structure of SVM concept classifiers and compare their performance with memory-based models, which lead to revealing observations. We propose ΔAP and ΔMAP as frequency-insensitive performance metrics. Finally, the heuristics for identifying reliable concept classifiers provide practical guidance as to which concepts are worthwhile to detect and use.

The remainder of the paper is organized as follows. Section 2 describes our experimental setup and our performance metrics. Section 3 describes the generalizability test on concept classifiers, and Section 4 explores the problem of how much these classifiers actually learn from training data. In Section 5, we identify the reliable concept classifiers and their common properties. We discuss the conclusions and future directions in Section 6.

2. EVALUATION SETTING AND PERFORMANCE METRIC

The experiments are carefully designed to be representative of the mainstream concept detection methods, so that the conclusions are general and convincing. As discussed below, we choose the data, semantic concepts, features, and learning methods that are frequently used in the literature and/or representative among various options. On the other hand, we propose new performance metrics to overcome the limitations of the existing ones.

2.1 Experiment set-up

Data: The experiments are conducted on two video collections used in the TREC Video Retrieval Evaluation [11]: the development set of 2005 (TREC05) and the development set of 2007 (TREC07). The TREC05 collection contains 86 hours of broadcast news video from 6 channels, which are CNN, NBC, MSNBC, CCTV, NTDTV, and LBC. Among them, CCTV and NTDTV are in Chinese (Mandarin), LBC is in Arabic, while the others are in English. Due to editing styles, target audience, and other factors, each channel exhibits distinctive data characteristics. Based on the provided shot boundaries, this 86-hour footage has 61,901 video shots, which are relatively evenly distributed across the 6 channels. The TREC07 collection contains 50 hours of news magazine, science news, news reports, documentaries, educational programming, and archival video provided by the Netherlands Institute for Sound and Vision, which can be collectively described as documentary video. This collection has 21,532 shots, and there is no further partitioning into channels or sub-collections.

Semantic concepts: The labels of 39 semantic concepts are provided on all the shots of TREC05 as part of the Light Scale Concept Ontology for Multimedia (LSCOM-Lite) project [6]. The labels of the same concepts except three of them (thus totally 36 concepts) are also available on TREC07. These concepts belong to various types, such as objects (e.g., *Car*), scenes (e.g., *Sky*), semantic topics (e.g., *Military*), and human activities (e.g., *Meeting*). There is a large difference between concepts in terms of frequency, which is the ratio of relevant shots among all the shots. General concepts have frequencies around 50%, such as *Outdoor*, while rare ones have frequencies below 1%, such as *Prisoner*.

Features: Each video shot is represented by the middle frame in the temporal axis as its "keyframe". The keyframe is described by a 225-d color moment feature computed from 5×5 grids and a 48-d Gabor texture feature. We concatenate them into a 273-d feature vector representing the video shot. In several TRECVID evaluations (e.g., [3]), this frequently used feature has shown to provide performance on par with other state-of-the-art visual features. Given the time and space constraints, we choose to focus on this representative feature set and leave the investigation of other features (e.g., SIFT-like local features) to future work.

Method: Almost all the methods apply SVMs with radius-basis kernel function (RBF) to train concept classifiers due to its practical success. Even sophisticated methods are built on SVM concept classifiers as the building blocks, e.g., combining SVM classifiers built on different features and classifiers for related concepts [10, 13, 1, 8, 12, 15]. It is fair to say that training SVM concept classifiers is the foundation of concept detection, and their performance is an important indicator of the status of this area. To make sure

the results of our study are general, we train all the concept classifiers based on SVMs with RBF kernel using the widely-used LIBSVM package [2].

The model parameters, especially gamma in the RBF kernel, have shown to have a large impact on the performance and were heavily tuned in practice. In our study, we set the model parameters to values that lead to respectable performance on the same concepts and data in TRECVID 2005 evaluation [3]. They are not necessarily the parameters that achieve the highest AP or MAP. This is not a problem since our focus is on the consistency and generalizability of performance. We care about the *change* of performance across different domains instead of the absolute performance in one domain. In fact, more tweaking of parameters may result in further overfitting and even worse generalization ability.

Setting: Concept classifiers are trained either on one of the 6 channels in TREC05, such as CNN, or on the entire collection, i.e, TREC05 or TREC07. The reason to have them trained from multiple datasets is to make conclusions more convincing and less likely the result of pure chance. This also allows us to test the generalization ability of concept classifiers across different channels and across different collections.

2.2 Performance metric: △AP and △MAP

Average precision (AP) is a standard performance metric of a concept classifier. Given the classifier's output as relevance scores on a set of shots, we rank the shots in descending order of their score, and compute AP as the average of the precisions of this ranked list truncated at each of the relevant shots. The mean of APs, or *mean average precision* (MAP), is a metric of the average performance of multiple concept classifiers.

While AP is a good metric of rank quality, as the metric of classifier performance it can be incomplete and misleading. The lower bound (baseline) of AP on a concept should be the AP of a "random classifier" that sorts the test shots *completely randomly*. Given the definition of AP, it is easy to see that the baseline AP of a concept is not zero; instead, it is equal to the frequency of that concept. Therefore, different concepts have different baselines, and the baselines have no relation with how well concept classifiers perform. For comparing classifiers, AP is misleading because it contains the baseline AP, giving concepts with higher baseline unfair advantage to concepts with lower baseline.

Two specific problems arise with the use of AP. First, it makes concept classifiers less comparable. For example, an *Outdoor* classifier with 0.9 AP is not necessarily better than a *Studio* classifier with 0.8 AP, as the random baseline (frequency) of the latter is much lower. Even for the same concept, the classifiers built on different collections are still not comparable because the concept's frequency varies with collections. The second problem comes with using MAP as the metric of average performance on multiple concepts, where MAP is the mean of the APs on these concepts. Because concept frequency varies greatly, sometimes by orders of magnitude, MAP can be easily dominated by the AP of a concept with a much higher frequency than the rest. On the other hand, a rare concept whose positive instances occur in nearly identical form (e.g., commercials) typically has a very high AP if one such instance is in the training data. This results in a large but somewhat deceptive improvement on MAP, without any generalizability to other domains.

		Training					
		CCTV	CNN	LBC	MSNBC	NBC	NTDTV
Test	CCTV	**.332**	.167	.172	.166	.172	.138
	CNN	.145	**.319**	.151	.193	.168	.132
	LBC	.161	.152	**.306**	.159	.180	.159
	MSNBC	.123	.176	.129	**.312**	.179	.122
	NBC	.140	.147	.152	.180	**.329**	.138
	NTDTV	.127	.129	.165	.141	.150	**.313**

(a) **MAP**

		Training					
		CCTV	CNN	LBC	MSNBC	NBC	NTDTV
Test	CCTV	**.256**	.088	.093	.089	.095	.061
	CNN	.075	**.250**	.080	.124	.100	.062
	LBC	.080	.070	**.226**	.079	.101	.077
	MSNBC	.058	.111	.064	**.249**	.116	.057
	NBC	.074	.081	.086	.116	**.265**	.072
	NTDTV	.058	.059	.094	.072	.081	**.242**

(b) **△MAP**

Table 1: Average concept detection performance as (a) MAP and (b) △MAP of 39 concepts with training and test data from any two of the 6 channels in the TREC05 collection.

A concept classifier should be evaluated as to how much *better* it performs than a random classifier. Thus, we define *delta AP* (or ΔAP) as the difference between the AP of a concept classifier and the random baseline of this concept in a particular dataset. In other words, ΔAP measures the *improvement* of rank quality as the result of using a classifier, which is a more faithful metric than AP for evaluating classifiers. While not directly measuring rank quality, a higher ΔAP definitely means better rank because $AP > \Delta AP$. Note that ΔAP can occasionally be negative if the classifier performs worse than random. We also define ΔMAP as the mean of multiple ΔAPs.

3. GENERALIZABILITY TEST

To see how well concept classifiers generalize, we build classifiers from one domain and compare their performance on the data from the same domain (i.e., within-domain performance) and from other domains (i.e., cross-domain performance). The experiment consists of two settings. In the *cross-channel* setting, we apply concept classifiers trained from one news channel of TRECV05DEV to another channel. In the *cross-genre* setting, we apply concept classifiers trained from the entire TREC05 to TREC07, or the other way around. The second setting is perhaps more challenging since it is between news video and documentary video, while the former is between news video from different broadcasters.

3.1 Cross-channel performance

For each of the 6 channels in TREC05, we build concept classifiers of the 39 LSCOM-Lite concepts using *all* the video data in that channel. This results in a total of 6×39 classifiers. The channel used for training a classifier is called *training channel*, while the channel to which the classifier is applied is called *test channel*.

Table 1 shows the average performance as MAP and ΔMAP of the concept classifiers for the 39 LSCOM-Lite concepts

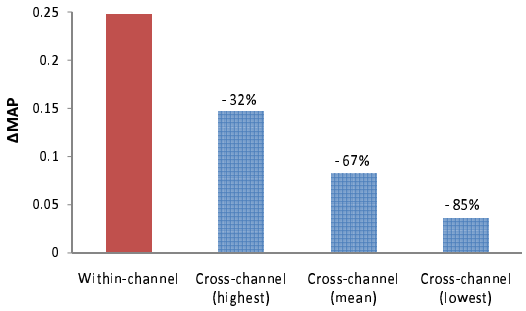


Figure 1: Comparison of within-channel performance and the highest, mean, and lowest cross-channel performance in ΔMAP.

under various training and test channels in two matrices. In each matrix, the numbers on the diagonal denotes *within-channel performance*, namely the performance when the training and test channel are the same. This performance is evaluated using 5-fold cross validation on each channel in order to avoid overfitting caused by using the same data for both training and testing. The numbers off the diagonal in each matrix denote the *cross-channel performance* of concept classifiers trained from one channel applied on a different channel.

It is clear from Table 1 that cross-channel performance of concept classifiers is *consistently* and *substantially* lower than the within-channel performance. While all within-channel MAPs are above 0.3, all the cross-channel MAPs are below 0.2, so the average decline is around 50%. This is a significant drop, but it is still biased by the random baseline, which is about 0.07 MAP as the average of 39 concepts. ΔMAP corrects this bias by factoring out the baselines. In terms of ΔMAP, the performance has a more significant decline at about 70%, from around 0.25 to below 0.1 in most cases. The performance decline is also universal, with the smallest decline around 50%, which happens when we apply MSNBC classifiers on CNN. While performance decline is expected, the severity of decline is surprising given that all the video data are of the same genre, i.e., broadcast news video. Overall, concept classifiers generalize poorly to channels other than their training channel.

There are two more interesting observations in Table 1. First, the channel with the highest within-domain MAP is CCTV with 0.332 MAP, and the channel with the lowest within-domain MAP is LBC at 0.306. However, the average cross-channel performance of CCTV classifiers is 0.139 MAP, lower than the average cross-channel performance of LBC classifiers, which is 0.151 MAP. This suggests that concept classifiers with higher performance in their training domain do not guarantee higher performance in other domains. In fact, they may have worse cross-domain performance as the result of overfitting. This warns against the common practice of heavy parameter tuning through cross validation in a single domain.

Second, the cross-channel performance of concept classifiers varies significantly with different test channels. As shown in Table 1, when applying classifiers trained from a fixed channel to the other 5 channels, there is a large gap between the highest and the lowest performance. For example, MSNBC-based classifiers achieve 0.193 MAP on CNN but only 0.141 MAP on NTDTV. Note that the basic assumption of supervised classification is that the training and test data follow the same distribution. Thus, the cross-channel performance is determined by how similar the training and test channel is in terms of data distribution. In the above example, MSNBC is definitely more similar to CNN than to NTDTV.

		MAP		Δ MAP	
	Training	TREC05	TREC07	TREC05	TREC07
Test	TREC05	0.294	0.143	0.223	0.073
	TREC07	0.166	0.201	0.086	0.122

Table 2: Average concept detection performance on 36 concepts with training and test data from TREC05 (news video) and TREC07 (documentary).

Figure 1 shows the within-channel performance and the highest, mean, and lowest cross-channel performance in terms of ΔMAP, where each performance is the average of the results on 39 concepts and 6 training channels. It is clear that the relation between training and test channel has a huge impact on performance. On average, concept classifiers lose only 40% performance when they are applied to the most similar test channels, but almost the entire performance (85%) on the most dissimilar ones.

3.2 Cross-genre performance

The cross-genre experiment involves both the news video in TREC05 and the documentary video in TREC07. To obtain the cross-genre performance, we build SVM classifiers for the 36 common concepts using all the data in either of the two collections, and evaluate them on the other collection. Similarly, the within-genre performance is obtained by 5-fold cross validation on each collection.

Table 2 shows the within-genre and cross-genre performance in terms of MAP and ΔMAP. Similar to the observation in cross-channel experiment, the cross-genre performance is significantly lower than the within-genre performance. The relative drop of ΔMAP is 61% when applying classifiers trained from news video (TREC05) to documentary video (TREC07), or 40% the other way around. Somewhat surprisingly, this performance drop is not larger than in the cross-channel setting, although the training and test data from the two genres are more dissimilar. A possible explanation is that the concept classifiers are built on a much larger training set (the data in all 6 channels are aggregated for training) and are therefore more reliable.

3.3 Discussion

The experiments undoubtedly show that concept classifiers generalize poorly beyond their own domain. The problem exists even between data that we might think are very similar, such as the news video from MSNBC and from NBC. This implies that existing concept classifiers offer little help on future data. Whenever a new domain emerges, one has to build new classifiers from scratch, which involves a huge amount of labeling effort and computational resource. For example, in TRECVID a large set of training data were manually labeled to build concept classifiers for test data of approximately the same size, and this effort was repeated every year. This practice simply does not scale in the face of an increasing variety of multimedia data.

An obvious reason for the poor generalizability is the difference on data distribution between domains, which violates the basic assumption of supervised learning. For example, a *Studio* shot may look very different in CNN and in CCTV, causing the classifiers to fail. But this is no excuse. As speech recognition systems should work with different speakers, concept detection should work consistently despite the source of the data, and ideally, despite the genre of the data. The real problem might be that our features are superficial and not domain-invariant, or that our classification algorithms are not smart enough to learn the essential patterns of each concept. We explore the question of how much our classifiers have learned from data in the next section.

4. LEARNING, OR MEMORIZATION

We approach the problem of whether our concept classifiers actually learn from the data or merely memorize it from two aspects. We first examine the ratio of data used as support vectors in these SVM classifiers, and then compare their performance with a memory-based approach.

4.1 Ratio of Support Vectors

The decision function of a SVM classifier is expressed as:

$$f(x) = \sum_{x_i \in D_{SV}} \alpha_i y_i \mathcal{K}(x, x_i) \quad (1)$$

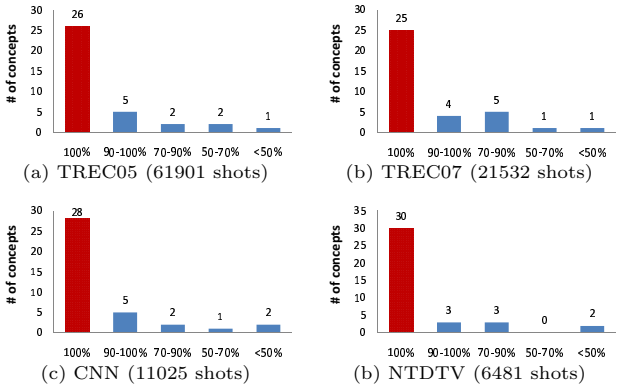
where D_{SV} is a subset of training data called support vectors (SVs), x_i and $y_i \in \{-1, 1\}$ are a SV and its label indicating its relevance to a given concept, and $\mathcal{K}(x, x_i)$ is the kernel function determining the similarity between the query point x and each SV x_i. This shows that the decision boundary of a SVM classifier is completely determined by the SVs, which are representative instances chosen by the SVM algorithm to define how the positive and negative data are separated. For example, training instances close to the decision boundary are typically chosen as SVs.

When RBF is the kernel function, the number of SVs indicates the complexity of the decision boundary and, as we will show, how much the SVM classifier has learned from the data. Think SVM learning as data compression. If the SVM algorithm finds a smoothed boundary to separate the two classes, it would need only a small percentage of data as SVs to represent that boundary. In this case, it has compressed the data into a small set of SVs while retaining the information about classification. If a smoothed boundary cannot be found, the SVM algorithm would produce a convoluted boundary that zigzags to place every positive instance on one side and negative instance on the other side. Such as boundary needs to be supported locally by a large number of SVs. In this case, it fails to compress the data. It does not do much learning besides saying "the neighborhood around positive data is positive, and the neighborhood around negative data is negative".

In Table 3, we show the ratio of SVs among the entire training data, among the positive data, and among the negative data, in SVM classifiers trained on the 6 channels and the TREC05 and TREC07 collection. All the ratios are averaged among the classifiers for the 39 concepts (36 for TREC07). The number of SVs in each classifier is obtained by examining the SVM model file generated by LIBSVM. We see that while the SVM classifiers do "compress" the data into a small set of SVs about only 12% to 15% of the training data, they retain most of the positive instances as

		Percentage of SVs in		
Dataset	**# shots**	**pos data**	**neg data**	**all data**
CCTV	10896	92.1%	10.5%	13.2%
CNN	11025	93.8%	9.3%	12.0%
LBC	15272	94.0%	10.4%	12.8%
MSNBC	8905	96.2%	9.7%	12.2%
NBC	9322	95.6%	11.4%	14.5%
NTDTV	6481	94.1%	10.9%	13.6%
TREC05	61901	94.7%	9.8%	12.4%
TREC07	21532	94.3%	12.6%	15.5%

Table 3: The average ratio of SVs in all data, in positive data, and in negative data of the SVM concept classifiers for the 39 concepts (36 for TREC07) trained on datasets of different sizes.



(a) TREC05 (61901 shots) (b) TREC07 (21532 shots)
(c) CNN (11025 shots) (b) NTDTV (6481 shots)

Figure 2: The number of concepts with ratio of SVs in positive data in different ranges on 4 datasets.

SVs. The compression is mainly from the negative data, which are abundant and less valuable to classification. The SVM classifiers fail to summarize the positive data, which are more valuable due to the imbalance of the two classes, into general and concise representations. As a result, they have to memorize most of them.

Figure 2 shows the distribution histogram of concepts against the ratio of SVs in positive data on 4 of the 8 data sets listed in Table 3. The distribution is very similar across the 4 datasets whose sizes are very different. For a majority of concepts, *all* the positive instances are used as SVs, and for many of the remaining ones, over 90% of the positive instances are SVs. Only on very few concepts is the ratio of SVs in positive data around or below 50%.

Overall, SVM concept classifiers do not learn the "general patterns" of positive data for most concepts, while they summarize negative data well. One explanation is the imbalance between the positive and negative data. However, we see no connections between the *absolute* number of positive data and the ratio of SVs in them. As shown in Table 3, from NTDTV to TREC05 the data size (and the size of positive data) increases almost 10 times, but the ratio of SVs in positive data remains in the small range of 92% to 96%. The *relative* ratio between positive and negative data (which is related to concept frequency) is a better reason, but there are still exceptions. The classifiers of some infrequent concepts use only a small ratio of positive data as SVs. For example, *Studio* classifiers use only 40.6% of the positive data as SVs, despite that only 10% of all the data

are positive. In contrast, classifiers for *Sky* which has a similar frequency (11%) use 97.3% positive data as SVs. This is because *Studio* shots are visually similar and can be well represented by a subset of them, which is not the case with *Sky*. So it seems that the high ratio of SVs in positive data is more related to the intrinsic property of each concept, such as irregularity of the distribution of positive data in the feature space, which in turn is related to the limitation of the visual features we used.

4.2 SVMs vs. kNN

From another perspective, we can view each term $y_i\mathcal{K}(x, x_i)$ in the summation of Eq.(1) as an atomic classifier that predicts the label of x to be the same as the label of SV x_i with confidence weighted by their similarity (distance). Therefore, a SVM classifier can be viewed as an ensemble of many such atomic classifiers, and its prediction the result of weighted majority voting by all the SVs. This shows that SVMs are similar to memory-based models such as k-nearest neighbor (kNN), except that they use only SVs rather than all the training data for prediction.

A memory-based model like kNN needs no training *pre se*; it merely memorizes the training data. The high ratio of SVs in positive data in the SVM concept classifiers implies that they too learn little beyond memorizing most of the positive data and some negative data. It is now interesting to compare the performance of the two in order to see whether SVMs perform any better than kNN. We use a kNN classifier that predicts a score for each query point x_q as the normalized weighted sum of the labels $y_i \in \{0, 1\}$ of its K nearest neighbor $x_1, ..., x_K$:

$$f(x) = \frac{\sum_{i=1}^{K} w_i y_i}{\sum_{i=1}^{K} w_i} \quad (2)$$

where the weight is set to the inverse of the Euclidean distance between x and each of its neighbors, i.e., $w_i = 1/D(x_i, x)$. We choose $K = 100$ based on preliminary experiments. Except for the denominator, Eq.(2) has a similar form to SVMs' decision function in Eq.(1) if we view the kernel $\mathcal{K}(x, x_i)$ also as a weight related to the similarity (distance) between x and x_i. The difference is that kNN makes predictions based on nearest neighbors while SVMs make predictions based on SVs. When most positive data are used as SVs, as shown above, a SVM classifier behaves very close to a kNN classifier with a very large K.

Figure 3 compares the (within-domain) performance of SVM and kNN concept classifiers side by side on each of the 6 channels and the TREC05 and TREC07 collection. The performance is measured by MAP of 5-fold cross-validation and is averaged from the 39 concepts (36 for TREC07). Surprisingly, SVM and kNN classifiers have comparable performance, despite the fact that the training-less kNN simply memorizes all the training data. One may argue that with careful parameter tuning SVMs may outperform kNN. While this is possible, we may also fine-tune kNN by varying K and the ways weight w_i is computed. Given that this comparison is done over many data sets, we believe SVMs and kNN are at the same level of performance in the case of concept detection. Together with the observation on the number of SVs, we see that SVM classifiers have "downgraded" themselves to memory-based methods in terms of both structure and performance. This is probably due to (again) the limitation of the visual features we used, and it is an interesting future work to see if the same observation occurs on other features.

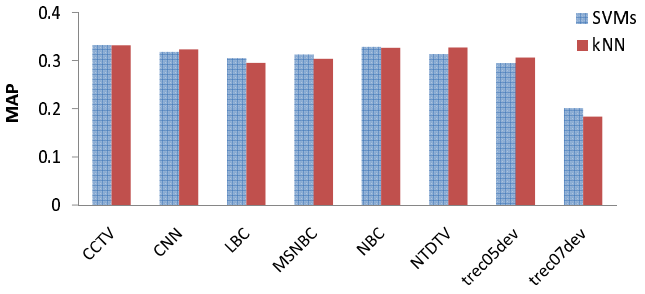


Figure 3: Comparison of the performance of SVM and kNN concept classifiers on different datasets. The performance is measured as MAP of 5-fold cross-validation and the average of 39 concepts (36 on TREC07).

At the same performance level, SVMs not only require longer training time than the training-less kNN, they are not necessarily faster in prediction. Given the form of its decision function Eq.(1), the prediction time of a SVM classifier increases linearly with the number of SVs. In comparison, there are efficient implementations of kNN which can significantly reduce the instances to be traversed in search for nearest neighbors. In fact, the ANN library for approximate kNN [5] we used in our experiment performs faster than LIBSVM in terms of prediction time.

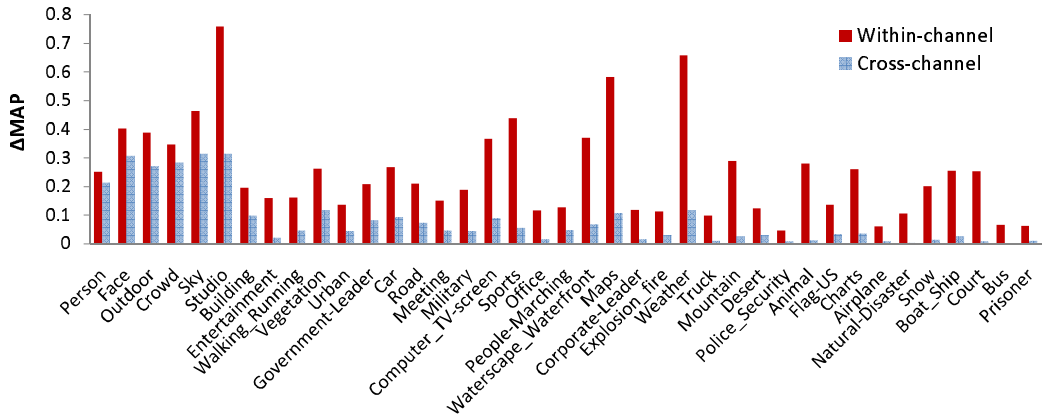
4.3 Discussion

The observation that SVM concept classifiers learn little besides memorizing the training data may help explain their poor generalizability. Because they memorize most of the positive data, they perform nearest-neighbor type of prediction, i.e., classifying data close to the positive SVs as positive and so on. This approach is fragile and it fails easily when the new data are distributed differently. For example, there are some gray-scale keyframes in TREC07 which distribute away from the color keyframes in TREC05 in the space of color features. As a result, concept classifiers trained on TREC05 perform poorly on TREC07. A concept classifier may generalize better if its boundary is more general and smoothed.

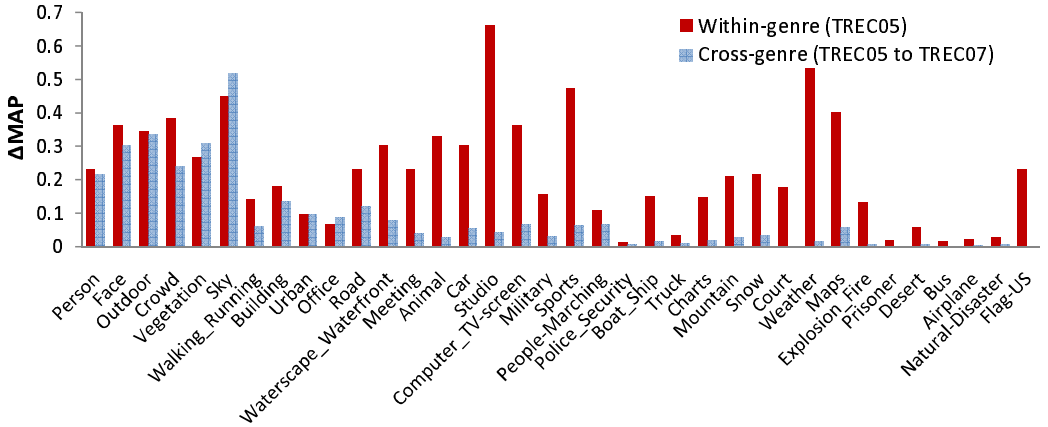
5. FINDING RELIABLE CONCEPT CLASSIFIERS

We have learned that in general concept classifiers learn little from data and generalize poorly. On a per-concept basis, however, classifiers of some concepts are more reliable than the others. It is important to find out these reliable concept classifiers so that we can use them with trust on other domains or in tasks such as retrieval. More important is to identify the common properties of reliable concept classifiers, which can guide us to discover new classifiers that are reliable *without* testing their generalizability on other domains. In this section, we explore the problem of what the reliable concept classifiers are and how to find them.

Figure 4(a) shows the within- and cross-channel performance as ΔMAP on 39 concepts ordered from left to right in descending concept frequency, where the performance is



(a) Within-channel performance vs. cross-channel performance



(b) Within-genre performance vs. cross-genre performance

Figure 4: The within- and cross-domain performance for the 39 concepts in descending order of frequency in (a) cross-channel setting and (b) cross-genre setting.

averaged from all training-test configurations. Figure 4(b) shows the within- and cross-genre performance as ΔAP on 36 concepts as we apply classifies trained on TREC05 to itself (via cross-validation) and to TREC07. The use of ΔAP and ΔMAP removes the impact from the difference on concept frequency and allows a realistic measure of the performance decline on each concept.

From Figure 4, we see a large variation between concepts in terms of the reliability of performance. The decline from within- to cross-domain performance is small for some concepts, such as *Person* and *Crowd*, but substantial for the others, such as *Animal* and *Charts*. Overall, reliable concept classifiers are more scarce than unreliable ones. Moreover, concept classifiers with higher within-domain performance are not necessarily more reliable, e.g., *Sports*. This shows (again) that optimizing for the performance on the training domain does not lead to more reliable concept classifiers.

A closer examination shows that the reliability of concept classifiers tend to increase as concept frequency increases. In fact, the 5 most frequent concepts have the smallest relative performance decline in cross-domain setting, while all the 19 concepts with frequency below 0.02 suffers a performance decline over 70%. So we speculate that reliable concept classifiers share common properties like high concept frequency and perhaps more.

We measure reliability by the *relative* decline from within-domain ΔMAP to cross-domain ΔMAP:

$$Decline = \frac{\Delta MAP_{within} - \Delta MAP_{cross}}{\Delta MAP_{within}} \quad (3)$$

where smaller decline indicates better reliability, and vice versa. To explore the common properties of concept classifiers, we plot in Figure 5 the distribution of relative decline from within- to cross-channel performance on each of the 39 concepts against (a) the concept frequency, (b) the average number of SVs, (c) the average number of SVs in positive data, and (d) the average ratio of SVs in positive data in the classifiers of the concept trained from 6 different channels. We also calculate the correlation coefficient between the decline and each factor, which is shown in the figure.

Figure 5(a) shows that the classifiers of frequent concepts indeed have smaller performance decline, which means they are more reliable than the classifiers of rare concepts. This trend is quite pronounced as indicated by a large (negative) correlation coefficient of -0.79. One reason is that frequent concepts are also generic ones whose definitions are relatively insensitive to domain changes (e.g., *Outdoor*, *Person*). Another reason can be that frequent concepts have a large number of positive data, which are necessary for building reliable classifiers, while rare ones do not have sufficient positive data.

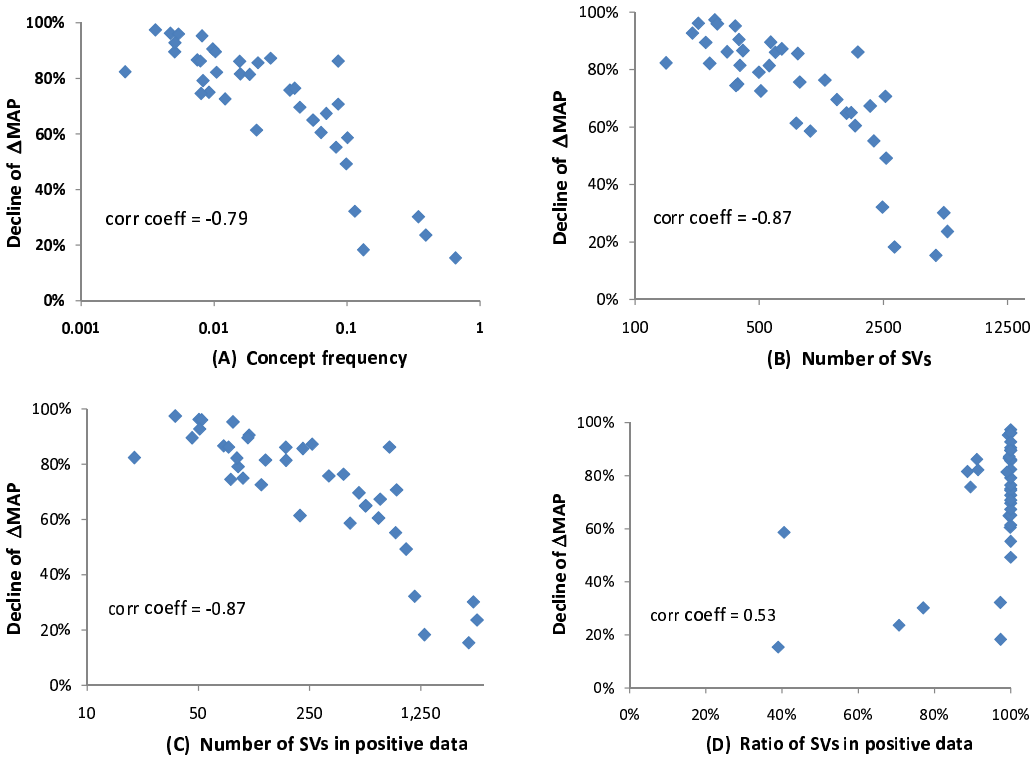


Figure 5: The relative decline of cross-channel ΔMAP from within-channel ΔMAP on each concept against (a) the concept frequency, (b) the average number of SVs, (c) the average number of SVs in positive data, and (d) the average ratio of SVs in positive data in the classifiers of the concept. The correlation coefficient between the decline on ΔMAP and each factor is shown in each figure.

The second point is somewhat confirmed by Figure 5(b) and (c), which shows that the performance decline is smaller when the total number of SVs or the number of SVs in positive data (i.e., positive SVs) in the classifiers increases. The correlation coefficients in both cases are -0.87, indicating an even stronger (negative) correlation than that between decline and concept frequency. Thus, more complicated and "bloated" concept classifiers tend to generalize better than simpler ones. This observation seems to be against the Occam's razor, which states that the simplest solution is the best. This is of no surprise, however, given our observation that SVM concept classifiers memorize most of the positive data and behave close to memory-based models. For memory-based models like kNN, a larger set of training data reduces model variation and usually leads to better performance. This helps explain why SVM classifiers bloated with a large number of SVs are more reliable.

Less obvious but more interesting is the correlation between performance decline and the ratio of SVs in positive data shown in Figure 5(d). While this ratio is 100% for many concepts, from the remaining ones we find that the performance tends to be more reliable if a concept classifier uses a smaller percentage of positive data as SVs. For the 4 most reliable concept classifiers, the ratio of SVs in positive data are 39%, 71%, 77%, and 97%, much lower than the average. Because the ratio of SVs in positive data indicates how much SVM classifiers have learned from the data, this observation implies that the more a SVM classifier learns from data, the better it will generalize to other domains.

On the one hand, concept classifiers with more SVs in positive data are more reliable. On the other hand, concept classifiers with a smaller ratio of SVs in positive data are more reliable. There is no contradictions between the two seemingly conflicting observations. If a SVM classifier is able to learn the pattern of training data, it needs only a small percentage of positive instances as SVs and still performs reliably. If not, it has to use most positive instances and resort to a memory-based approach for classification. In this case, more positive SVs help reduce model variation and yield better performance.

To confirm our findings, we repeat the same analysis in the cross-genre setting by applying concept classifiers trained on TREC05 to TREC07 and analyzing the performance decline on a per-concept basis. In this case, the concept classifiers are trained with a much larger set of training data. Figure 6(a) - (d) shows the distribution of performance decline against the concept frequency, the number of SVs, and the number and ratio of SVs in positive data in each concept classifier. While the correlations are slightly weaker, the general trend is consistent with that in the cross-channel experiment. Classifiers of frequent concepts and bloated classifiers with a large number of SVs appear to be more reliable across different genres (collections).

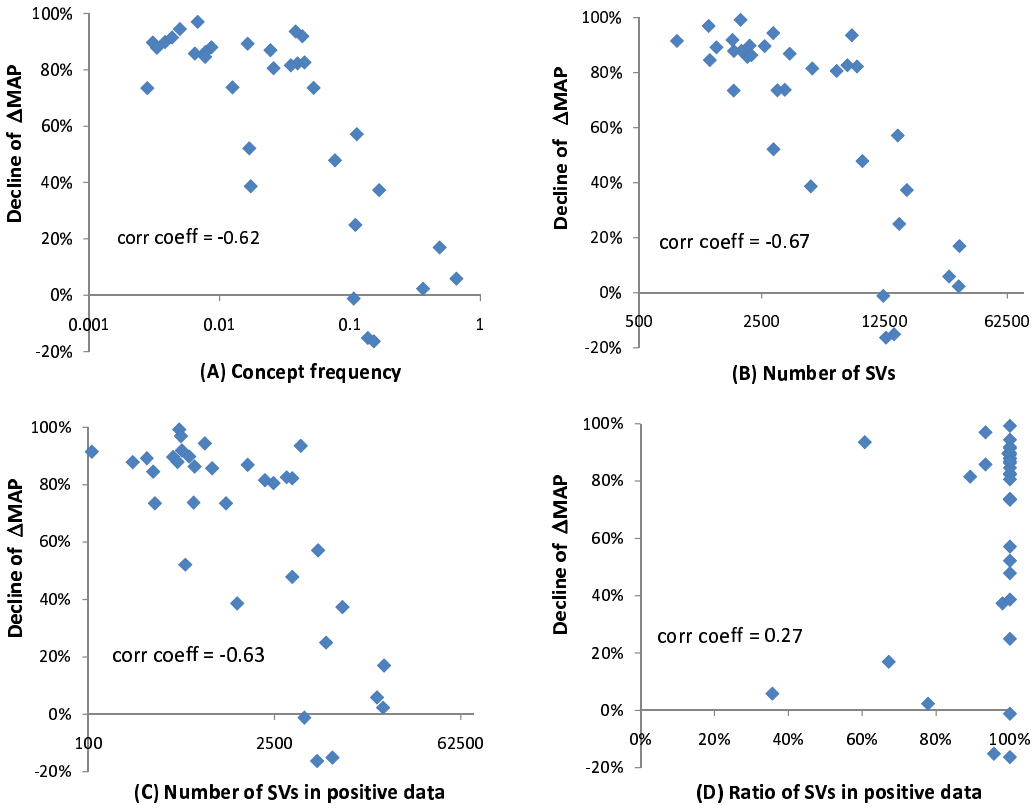


Figure 6: The relative decline of ΔMAP on each concept against (a) the concept frequency, (b) the number of SVs, (c) the number of SVs in positive data, and (d) the ratio of SVs in positive data in the concept classifier, when applying concept classifier trained from TREC05 to TREC07.

6. CONCLUSION AND FUTURE WORK

We summarize the highlights of our findings as follows:

- Concept classifiers trained using the standard approach are generally *unreliable* because they generalize poorly to domains other than it training domain, indicated by significant and consistent decline of performance.
- The SVM-based concept classifiers are generally unable to learn much from data and have to retain most of positive data as support vectors. This makes them behave close to memory-based approaches such as kNN, which is confirmed by the comparable performance between the two models.
- There is a large variation in terms of reliability among concept classifiers. Classifiers of frequent concepts, classifiers bloated with many SVs, and classifiers using a small percentage of positive data as SVs, tend to be more reliable than the others. This helps us identify reliable concept classifiers *a priori* without having to evaluate their generalizability to all possible domains through expensive experiments.

These observations cause serious concern. Among nearly 40 concepts, all except a precious handful do not have generalizable classifiers. Ungeneralizable classifiers are of limited use in the face of a growing variety of multimedia data, and building new classifiers for every domain is a daunting task in terms of annotation effort and computation. Moreover, our observations are limited because the experiments focus on data with similar nature, i.e., news video from different channels and documentary. As an area of future work, we plan to study the generalizability issue between data of different nature, such as news video and Web video (e.g., YouTube video), where the problem can be more serious.

Semantic concepts were proposed to bridge the gap between low-level features in image and video data and high-level user queries. We argue that these concepts themselves constitute too large a gap to the low-level features. On the feature side, this paper has shown that global features on the distribution of color, texture, or edge are too limiting and data-sensitive to capture the essential patterns of most concepts, resulting in poor generalizability across domains. Local or keypoint-based features (e.g., SIFT, bag-of-features) have shown promising results in image and video classification [9, 8]. We plan to study their reliability as interesting future work. Moreover, it is beneficial to use specialized features and approaches for certain concepts. A successful example is the use of a face detector for the *Face* concept. One specialized approach may detect only one concept, but in a more reliable way. After all, it is quality instead of quantity that really matters.

On the concept side, many concepts are simply too high-level to be detected from the features available. It is hard to imagine that concepts requiring high-level human judgement, such as *Prisoner*, *Corporate-leader*, can be reliably detected from color and texture features. The (deceptive)

performance on them is achieved by recognizing contextual clues such as background, which are seriously data dependant. Concepts like *Outdoor* and *Sky* have better correspondence with low-level features and as our study shows, they are indeed more reliable (and deservedly so). This shows that we should focus on less "semantic", intermediate-level concepts that have an explainable, demonstrated connection with low-level features, such as *Desert* and *Waterscape*. Given the complexity of many video scenes, we may also consider training concept classifiers based on image regions rather than the whole images or video shots.

Another future work is on adapting concept classifiers from one domain to another. Several approaches have been proposed on this direction. For example, cross-domain SVM (CDSVM) proposed in [4] updates existing classifiers to a new domain by re-training over the previously learned support vectors and training instances from the new domain. In comparison, adaptive SVM (aSVM) proposed in [14] directly modifies the decision functions of the existing classifiers based on the limited training instances from the new domain. These approaches improve the performance on new domains at the cost extra labeling, but they do not make concept classifiers fundamentally more reliable. Adapting existing classifiers is meaningful only when they provide useful information on the new domains. With the observations in this study, we can make a better judgement as to which concept classifiers to adapt and which to discard (and replace with new classifiers).

7. REFERENCES

[1] M. Campbell, A. Haubold, S. Ebadollahi, M. Naphade, A. Natsev, J. Smith, J. Tesic, and L. Xie. IBM Research TRECVID-2006 Video Retrieval System. *TREC Video Retrieval Evaluation Proceedings*, 2006.

[2] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001.

[3] S. Chang, W. Hsu, L. Kennedy, L. Xie, A. Yanagawa, E. Zavesky, and D. Zhang. Columbia University TRECVID-2005 Video Search and High-Level Feature Extraction. *TREC Video Retrieval Evaluation Proceedings*, 2005.

[4] S. Chang, W. Jiang, A. Yanagawa, and E. Zavesky. Columbia University TRECVID 2007 High-Level Feature Extraction. *TREC Video Retrieval Evaluation Proceedings*, 2007.

[5] D. M. Mount and S. Arya. ANN: A Library for Approximate Nearest Neighbor Searching.

[6] M. R. Naphade, L. Kennedy, J. R. Kender, S. F. Chang, J. Smith, P. Over, and A. Hauptmann. A light scale concept ontology for multimedia understanding for TRECVID 2005. In *IBM Research Technical Report*, 2005.

[7] M. R. Naphade, T. Kristjansson, B. Frey, and T. Huang. Probabilistic multimedia objects Multijects: A novel approach to video indexing and retrieval in multimedia systems. In *Proc. of ICIP*, 1998.

[8] C. Ngo, Y. Jiang, X. Wei, F. Wang, W. Zhao, H. Tan, and X. Wu. Experimenting VIREO-374: Bag-of-Visual-Words and Visual-Based Ontology for Semantic Video Indexing and Search. *TREC Video Retrieval Evaluation Proceedings*, 2007.

[9] J. Philbin, O. Chum, J. Sivic, V. Ferrari, M. Marin, A. Bosch, N. Apostolof, and A. Zisserman. Oxford TRECVid 2007 Notebook paper. *TREC Video Retrieval Evaluation Proceedings*, 2007.

[10] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, T. Mei, and H.-J. Zhang. Correlative multi-label video annotation. In *Proc. of the 15th ACM Int'l Conf. on Multimedia*, pages 17–26, 2007.

[11] A. Smeaton and P. Over. Trecvid: Benchmarking the effectiveness of infomration retrieval tasks on digital video. In *Proc. of Conf. on Image and Video Retrieval*, 2003.

[12] C. Snoek, I. Everts, J. van Gemert, J. Geusebroek, B. Huurnink, D. Koelma, M. van Liempt, O. de Rooij, K. van de Sande, and A. Smeulders. The MediaMill TRECVID 2007 Semantic Video Search Engine. *TREC Video Retrieval Evaluation Proceedings*, 2007.

[13] R. Yan, M. yu Chen, and A. G. Hauptmann. Mining relationship between video concepts using probabilistic graphical model. In *IEEE Int'l Conf. on Multimedia and Expo*, 2006.

[14] J. Yang, R. Yan, and A. Hauptmann. Cross-domain video concept detection using adaptive svms. *Proceedings of the 15th international conference on Multimedia*, pages 188–197, 2007.

[15] J. Yuan, Z. Guo, L. Lv, W. Wan, T. Zhang, D. Wang, X. Liu, C. Liu, S. Zhu, and D. Wang. THU and ICRC at TRECVID 2007. *TREC Video Retrieval Evaluation Proceedings*, 2007.