University of Kaiserslautern
Image Understanding and Pattern Recognition
Prof. Dr. Thomas Breuel

Master Thesis

Efficient Domain Adaptation for web-based Video Concept Detectors

Damian Borth

March, 2010

Examiner:
Prof. Dr. Thomas Breuel

Supervisor:
Dr. Adrian Ulges

Hiermit versichere ich, dass ich die vorliegende Masterarbeit selbständig verfasst und keine anderen als die angegebenen Hilfsmittel verwendet habe. Alle Textauszüge und Grafiken, die sinngemäß oder wörtlich aus veröffentlichten Schriften entnommen wurden, sind durch Referenzen gekennzeichnet.

Kaiserslautern, im März 2010

(Damian Borth)

Acknowledgments

I wish to express my thanks to all the people who helped and supported me with this thesis. First, I would like to thank Prof. Thomas Breuel for giving me the opportunity to work on this very exciting topic. I would like to thank Adrian Ulges for constantly providing support and guidance during the completion of this work and for his *colorful* feedback and insights into the art of scientific writing.

I also would like to thank everyone at MADM for not killing me when I was using 95% of the group's computational resources. Especially, Markus Goldstein and Christian Schulze who patiently explained to me that my experiments interfere with the group's backup plans. In this context I also would like to thank Markus Koch for keeping the machines running (because this is not only *rocket science*, it is...) and Tsvetana Spasova for her help related to the dataset acquisition. Also, I would like to thank Alexander Arimond for spending more than one evening with me at the office while writing on his thesis and discussing off-topics like Gödel, Hilbert Spaces and Kaiser Friedrich III. Further, I would like to acknowledge Joost van Beusekom's help in reading the acknowledgment section of this thesis. Finally, I would like to give a special thanks to Waldemar Borth and Marco Schreyer for the Boland's Saturday Morning Coffee Sessions, which are keeping me sane.

Abstract

In this thesis, the visual learning of automatic concept detectors from web video as available from services like YouTube is addressed. While allowing a much more efficient, flexible, and scalable concept learning compared to expert labels, web-based detectors are known to perform poorly when applied to different domains (such as specific TV channels). This *domain change problem* will be tackled by using a novel domain adaptation approach, which initially trains a source domain classifier on web video content and successively performs a highly efficient online adaptation on the target domain.

In quantitative experiments on data from YouTube and from the TRECVID campaign, first, the influence of domain change is quantified and validated to be the key problem for web-based concept learning, with a much more significant impact than other phenomena like label noise. Second, the proposed domain adaptation approach is shown to improve accuracy of web-based detectors significantly as also being comparable to the level of SVMs trained on the target domain. Finally, the approach is extended with active learning such that adaptation can be interleaved with manual annotation for an efficient exploration of novel target domains.

Zusammenfassung

Das Thema dieser Arbeit ist das visuelle Lernen von automatischen Konzept-Detektoren unter Verwendung von online Videomaterial, welches von Plattformen wie YouTube zur Verfügung gestellt wird. Diese Art des Lernens erlaubt ein viel effizienteres, flexiblereres und skalierbareres Konzeptlernen als jenes mit konventionellen, von Experten manuell annotierten Datensätzen. Als nachteilig erweist sich jedoch die Tatsache, dass web-basierende Detektoren an Präzision verlieren, wenn sie auf einer anderen Domäne angewendet werden als online Video (wie z.B. traditionelles Fernsehn). Diesem *Domain Change Problem* wird mit Hilfe einer neuartigen Adaptatiosnmethode entgegengewirkt, welche initial einen quellendomänenspezifischen Klassifikator auf online Videomaterial trainiert und anschließend sukzessiv eine effiziente Adaption auf der Zieldomäne durchführt.

In quantitativen Experimenten auf YouTube und TRECVID Daten wird zunächst gezeigt, dass der Wechsel der Domäne eine signifikante Auswirkung auf die Qualität von Videoklassifikatoren hat und dadurch ein Schlüsselproblem für das web-basierende Konzeptlernen darstellt, ein Umstand welcher weit signifikanter ist als schwach annotierte Trainingsdaten. Zusätzlich wird gezeigt, dass der vorgestellte Adaptionsansatz die Präzision von web-basierenden Detektoren weitreichend verbessern kann und diese sogar vergleichbar zu SVMs macht. Abschließend wird die vorgestellte Methode mit Ansätzen des *Active Learning* erweitert, so dass ein abwechselndes Adaptieren und manuelles Annotieren ermöglicht wird. Dieses führt zu einer viel effizienteren Exploration neuer Zieldomänen.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Currently, the media landscape experiences a major shift towards more distributed structures called *social media* or *citizen journalism.* Examples illustrating this are the Hudson River airplane crash or the public protest in Iran after the presidential election in 2009. One key element of triggering this is the upload and distribution of images and videos over the Internet minutes after such incidents happened.

Focusing on video, this trend is only one example of the increasing presence of digital video in our everyday life. Nowadays, we are used to generate large amounts of video material and to publish it online via video portals like “YouTube”, “Vimeo”, or “Blinkx”[1]. For example, YouTube, as the market leader, is storing about 20 hours of new video content per minute in its database [48] and is delivering over one billion videos to its users every day [116]. Additionally to online video portals, live-streaming services like “Ustream”, “Justin.tv” and “Livestream”[2] create another quickly growing area for digital video. Such companies offer video live streams which – combined with social networks – lead to a new form of interactive TV experience that’s already streamed in massive amounts over the web [42, 62]. Beside web video portals and live streaming services a third form of digital video is moving towards the Internet: traditional TV. Broadcasting networks like NBC and News Corp., for example, are embracing on demand video streaming with their premium content platform “Hulu”[3], which is nowadays already the second most visited video website in the US [96]. Other networks like HBO are either shortly before launching their online TV platform (“HBOGo”[4]) or extend their service with online distribution channels like in the “TV Everywhere” initiative from TimeWarner and Comcast [95]. Putting it together, there is not only an immense amount of digital video already stored but also a rapidly growing trend to make even larger quantities of video content available online. This is particularly reflected in a recent report by CISCO [41], where digital video is estimated to occupy around 91% of all Internet traffic by 2013.

However, making video content available online does not imply to make it searchable to the end user. While community efforts like collaborative tagging (also called *folksonomy*) are aiming to provide searchable annotations (or tags) [33], they are prone to spam [52], include misspellings, numbers [20] or are subjective or non-relevant [97]. Therefore, the lack of accurate and detailed metadata is still a major problem in video search, making it a cumbersome and frustrating task for end users.

[1] www.youtube.com, www.vimeo.com, www.blinkx.com

[2] www.ustream.com, www.justin.tv, www.livestream.com

[3] www.hulu.com

[4] www.hbogo.com

1.1 Concept-based Video Retrieval

In the literature two groups of access mechanisms to video databases can be found: "query-by-example" [3, 10, 25], where an image is given as an example and similar images are retrieved from the database, and "textual-search", where the user types a textual query and the retrieval system connects the given keywords with videos stored in the database. While the first query approach might favor a browsing alike exploration of the database, the latter one is considered to be a more natural querying mechanism to the end user [110] and therefore is used more in the context of folksonomy driven environments as well as regular search engines. One way to build a textual index is to take experts and let them label the database manually according to well defined semantic concepts describing objects ("airplance flying"), scene types ("cityscape") and activities ("person playing soccer") appearing in the videos. This, however, is infeasible due to the huge size of current video databases. This situation remindes the one of early web search, where Yahoo with its expert generated web index was challenged by Google and its automatic machine generated web index allowing Google's web search to easily scale-up to cover the entire visible web. Following, the question is if an indexing of video databases can also be done automatically, i.e. can a machine build such an index?

This challenging task is referred to as *concept detection* [85] and is subject to intensive research by a large research community [64]. Following, concept detection systems aim to infer the probability for predefined target concepts to appear in a given video clip. This is accomplished by treating concept presence as a binary classification problem under the usage of extracted low-level features like color, texture or motion from the video stream and statistical learning methods. The difficulty behind this can be summarized by the semantic gap [84], the mismatch between low-level features of a video stream on the one side and a user's high-level interpretation of the video on the other. Although the performance of state-of-the-art concept detection systems [89, 105, 111] is yet not as good as manual annotation [64, 82], the idea of automatically mining large video databases for semantic concepts is considered to be a key building block of video search prototypes [1, 83, 86, 85].

1.2 Visual Learning from Web Video

One particular problem of concept detection is its demand for labeled training sets, which serve as a foundation for supervised machine learning, the underlying technology of current concept detection systems. So far, training samples have been acquired manually, i.e. a human operator labels videos or video shots with respect to concept presence. Thereby, concepts are well defined according to a concept vocabulary [59]. This time-consuming and cost-intensive effort [5, 87, 113] indeed leads to high quality training material, but suffers from a scalability problem raising the question if alternative sources for concept detector training data exist.

Recently, the usage of socially tagged web images & video as alternative sources of training data for semantic concept detection has become more prominent [78, 98, 99, 101]. Utilizing such data offers the following advantages over a training from a small set of expert labels: first, it allows to learn large concept vocabularies which are required to cover users' information need and thus lead to more efficient search [38]. Second, it enables concept detection systems to be more flexible in learning new emerging concepts like "Vancouver Olympics 2010", "Haiti Earthquake" or "iPad" [5]. Third, it prevents overfitting as learning from only a small set of sample videos tends to deliver detectors that generalize poorly [112].

[5] top ranked searches 2010 by "Google Insights for Search" for web search, news search and product search respectively - as retrieved March. 2010

Web video is publicly available at large scale from online portals like YouTube, Vimeo or Blinkx and is associated with a noisy but rich corpus of tags, comments and ratings that are provided by large communities. Utilizing this information might replace expert labeled datasets by automatically harvesting training material from the web. For example, to learn a concept like "person playing soccer" a search query has to be formulated and sent to one of the previously mentioned web video portals. The resulting list of relevant videos can now be downloaded and used as training material. For this purpose tags are used as positive labels for concept learning. Web video has already been proven to train more general detectors performing better on unseen datasets as compared to detectors trained on specific expert labeled data [101] and demonstrated its potential as a comprehensive training source for visual concept learning [98].

Web Video Characteristics Web video, when used as training source for concept detection has its own characteristics. First, user tagged web video has – when compared to expert labeled material – a differently motivated labeling. Experts annotate videos according to well defined concept definitions and independent of their personal interest, whereas web users strongly follow the *focus of interest* [98] i.e. they prefer to tag objects in the main focus of the video and not taking care of concepts which might appear in the background. This slightly different annotation behavior is not directly leading to training material which is semantically wrong but might mislead concept learning in not providing the full range of visual clues for a given concept. Another circumstance to have in mind when using tag based services as source for concept detection training is the mapping of search queries to defined semantic concepts during training data download. Taking YouTube as an example, retrieval of training material is a two step process where first the search engine is queried by a set of keywords delivering a list of relevant video and second the download of those videos. The first steps is crucial for the success of concept detector training as a straightforward schema for such a mapping is not available and therefore usually must be done manually.

Challenges in Dealing with Web Video The usage of web video faces two challenges: first, web video is *weakly labeled* i.e. its annotations are often noisy, subjective, unreliable and coarse containing a great amount of non-relevant content. For example, at YouTube where tags can only be given at video level the fraction of non-relevant content lies between $50-80\%$ leading to a significant performance loss of web-based concept detectors when compared to ones trained on labels given by experts [97]. To deal with this *label noise problem*, several approaches have been proposed, either filtering non-relevant material [11, 97, 100, 107] or focusing on tag coarseness [32]. Second, when concept detectors are trained on web video and afterwards applied to a different source of video data (or *domain*) we face the so-called *domain change problem*: a significant discrepancy of the visual appearance between given domains. This challenge is the focus of this thesis and will be described in the next section in more detail.

1.3 The Domain Change Problem

One key problem with concept detectors is that they work well on the data they are trained on but generalize poorly to other data sources (or *domains*) [112]. In this context, the definition of a domain is the following [114]:

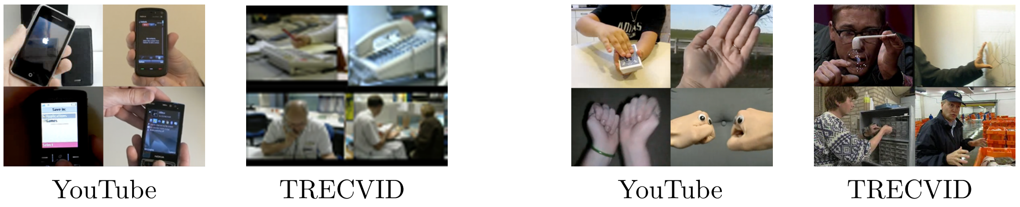


Figure 1.1: Frames from YouTube and TRECVID videos tagged with “Telephone” (left) and “Hand” (right). The domain change leads to a different visual appearance of both concepts.

> ***Domain:*** A domain refers to data of a certain type, from a certain source, or generated over a certain period in time.

Following, a domain can be a particular TV channel, a content provider, a video genre like documentary content or a period of time like the *film noir* movies created in the early 1940s to late 1950s. Particularly, web video also defines its own domain considering the definition above. In practice, concept detectors are often trained on one domain and applied to another one. This has been reported to degrade detector performance [101, 113], a phenomenon that is comonly refered to as *domain change problem* and is illustrated in Fig. (**1.1**): imagine training a detector for the concepts “Telephone” or “Hand” on web video (here, YouTube data), which shows mostly close-ups, and applying it on TV broadcast data (here, the TRECVID dataset [64]), which shows mostly office telephones or hands in interview scenes. Obviously, the web-based detectors will perform poorly on this particular dataset. This raises the question whether detectors trained on one domain (the “source domain”) can be *adapted* to another one (the “target domain”). So far, this challenge has been addressed using techniques like *cross-domain learning* or *domain adaptation* in the context of switching between different TV domains [28, 44, 113, 114].

1.4 Goal and Outline

This thesis addresses the training of concept detectors on web video and their application to different domains of video data. Particularly, the challenges resulting from the domain change problem are addressed. The goal of this work is to analyze the impact of the domain change between web video and specific target domains and to adapt web-based detectors by a novel domain adaptation technique as illustrated in Fig. (**1.2**): a first training on large-scale web material leads to an initial detector, which is then used as a starting point for a few cycles of adaptation utilizing labeled samples from the target domain. In the context of web-based concept learning, where we want to learn *many* concepts from *large-scale* training data, several new questions need to be answered:

1. Can social-tagged web video improve detectors compared to training only on domain-specific data?
2. How much does the associated label noise of web video affect domain adaptation?
3. Can we do an efficient, light-weight adaptation of web-based detectors to other domains (enabling adaptation in an “interactive search” fashion [78])?
4. How many samples from the target domain are needed for a successful adaptation?

This is why web video requires a seperate investigation regarding the well-stated problem of domain adaptation.

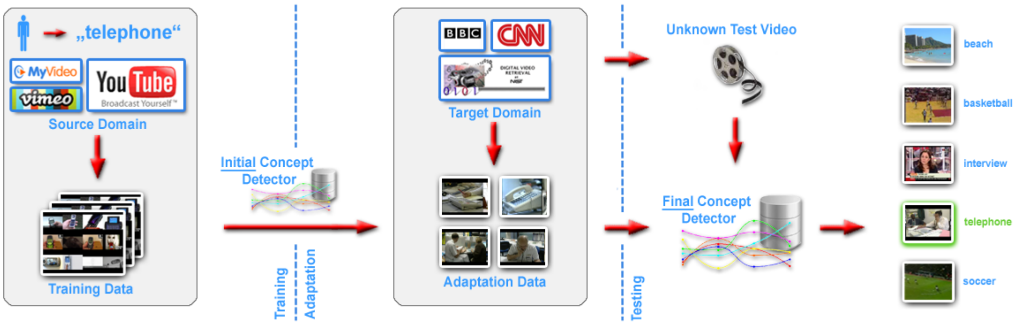


Figure 1.2: To learn a concept, the proposed system downloads videos from web video portals and trains an initial detector. This detector is adapted efficiently using few labeled samples from the target domain, obtaining a final domain-specific detector.

In detail, the contribution of this thesis are the following:

- **domain change impact**: first, the impact of domain changes on concept detectors is analyzed. This includes different supervised learning methods and combinations of training data. It will be demonstrated that domain changes degrade concept detection performance significantly.
- **adaptation**: a novel and efficient adaptation approach is proposed and demonstrated to lead to significant improvements – adapted detectors can outperform purely web-based ones and also the ones trained on the target domain.
- **adaptation with few samples**: as the acquisition of labeled training samples is cost-intensive, a minimal amount of samples should be required for adaption. For this purpose, active learning methods [5] are evaluated. These offer the advantage that user feedback can be incorporated into the domain adaptation framework.

The thesis is organized as follows: the second chapter covers work related to concept detection, domain adaptation and active learning. Chapter 3 outlines the proposed approach of an online learning domain adaptation technique whereas section four presents experimental results on real world video data. Finally, Chapter 5 provides a conclusion and discussion.

Chapter 2

Related Work

In this chapter, first supervised concept detection will be introduced. It will be continued with general domain adaptation techniques where classifiers are trained on particular source domains and are applied to different target domains. This will include approaches proposed in machine learning, information retrieval, data mining and *cross-domain learning*, the term that is used in the concept detection community. Finally, an overview of active learning will be given, a common approach to label and explore unknown datasets.

2.1 Concept Detection

Concept detection is the task of inferring semantic concepts in video streams. This is achieved by computing posterior probabilities which indicates the likelihood of presence of concepts from a vocabulary. A comprehensive overview of the field can be found in [85].

A major research effort in the context of concept detection is the TRECVID [81] benchmark, which addresses concept detection in its High-Level Feature Extraction task [82]. This benchmark aims to evaluate the performance of different concept detection systems on common, standardized datasets and allows the community to exchange experience and drive this area of research forward. The architecture of most concept detection systems as benchmarked in TRECVID can be described by five major blocks: pre-processing, features extraction, statistical classification, fusion, and concept relation modeling:

2.1.1 Pre-processing

A video consists of a sequence of shots being separated by cuts or gradual transitions. Since the basic unit of information in concept detection is a shot, the first step is to temporarily segment a video into its shots [54, 67]. From these keyframes are extracted, which serve as input for visual concept learning in the next processing steps. Here, either the middle frame of a shot may serve as a keyframe or a set of keyframes may be taken to represent the content of the associated shot [12, 37].

2.1.2 Feature Extraction

The purpose of feature extraction is to transform keyframes into n dimensional feature vectors $x \in \mathbb{R}^n$ which can be used during the subsequent classification step. Many different descriptors have been proposed, ranging from global color, texture and motion descriptors [27] to patch-based ones like the very often used *bag-of-visual word descriptor* [80] with SIFT [55] or SURF [6] features. Especially patch-based descriptors proofed to be robust and give high accuracy in several computer vision tasks [30, 45, 103]. For further information on feature extraction please refer to evaluations in [27, 103].

2.1.3 Statistical Classification

Given feature vectors $x \in \mathbb{R}^n$ the next processing step is the inference of concepts using statistical classification. A popular choice to achieve this are Support Vector Machines (SVMs) [76, 104], which are used in most systems nowadays [85]. SVMs, configured as binary classifiers, deliver scores which – transformed into posterior probabilities $P(t)$ [68] – indicate the absence or presence of a concept t. Alternative approaches include linear discriminative classifier [66], kernel dicriminant analysis [89] or neural network based mutitask learning [34]. Such supervised machine learning requires a labeled training set prior to classification. This training data can be acquired by an expert [5] or by alternative sources like web video [78, 98, 101].

2.1.4 Fusion

Many systems utilize multiple types of features and supervised learners [2, 36]. A fusion or combination of such components can be performed on two different levels: feature fusion [2, 36] i.e. the concatenation of feature vectors previous to detector learning, or classifier fusion where detector results are combined after classification [45, 88, 105]. While the former offers the advantage of utilizing feature dependencies the latter one does not have to deal with an increased high dimensional feature vector as in machine learning it is prefered to work in a low dimensional feature space due to the curse of dimensionality problem.

2.1.5 Concept Relation Modeling

One additional step is semantic relation modeling between concepts [60, 70, 89]. Here, the fact is exploited that the presence of a concept serves as an additional indicator for a related concept, e.g the presence of a “sky” indicates an increased probability of an “airplane” being present. Such co-occurrences or correlation between concepts can be modeled with different approaches. For more details please refer to [85].

The optimization of the final concept detector performance through the processing pipeline is now a question of selecting the best of all available paths through the system setup, which is usually achieved through the help of proper validation data.

2.2 Domain Adaptation

This section discusses domain adaptation from different points of view. A common setup is defined as follows: a supervised learner is trained initially on labeled samples $x_1, \ldots, x_n \wedge y_1, \ldots, y_n \in \{+1, -1\}$ from a source domain $\mathcal{D}^s$ and is employed on a specific target domain $\mathcal{D}^t$. In such a setup it is likely that the data distribution of the source and target domain are different and therefore the performance of the supervised learner will not be optimal. To reduce the effect of such a domain change, different techniques have been proposed (for a detailed survey refer to [43, 115]):

2.2.1 Domain Adaptation Approaches

Following a summary of domain adaptation approaches is given, which are not directly related to concept detection but cover other applications in machine learning.

Sample Bias Correction & Covariance Shift A first approach towards domain adaptation is *Sample Bias Correction* or *Covariate Shift.* This are two approaches, where the focus lies on dealing with different distributions of training and test data. Here, the basic assumption is that the expected loss on the test distribution equals a re-weighted expected loss on the training distribution. Transferring this idea to domain adaptation, Blitzer proposed an approach [8], where *pivot features* that appear frequently in both domains are selected by heuristic methods and are reweighed appropriately before training. A more theoretical investigation of this method can be found in [7]. Related approaches which additionally utilize the structure of unlabeled samples are proposed as covariate shift [91] or sample selection bias [40].

Feature Replication Another data manipulation technique is *Feature Replication* [24], where features are augmented for kernel function construction. The idea of feature replication is to combine source and target domain datasets to find common feature characteristics so that the source domain can provide new (replicated) data for the target domain. A disadvantage of this method is the increased dimensionality of the feature vector and the resulting increase of model complexity.

Transfer Learning Another area in machine learning similar to the domain adaptation setup is *Transfer Learning.* In transfer learning, knowledge from one related task (in case of multiple related tasks it is called *Multi-Task Learning*) is used to improve performance of a target task. One group of methods uses weighted auxiliary or prior data in combination with SVMs [108, 109], whereas another group of methods focus on unsupervised transfer learning [65, 71]. A more general review of the topic can be found in [23, 57, 58].

Incremental Learning An alternative to transfer learning is *Incremental Learning* or *Online Learning.* Here, classifiers are successively updated according to a continuous sample input and therefore can adapt naturally to a new domain given a sufficient number of samples from the target domain. In particular, the proposed approach in this thesis can be seen as an example of online learning adaptation. A detailed survey about this type of online learning is provided by [22]. Other examples of SVM-based methods in this area can be found in [13, 49].

Drifting Concept Detection Another field close to domain adaptation is *Drifting Concept Detection* known from the data mining community. Here, however, a drifting concept is a statistical property of a predicted target variable which changes over time due to the change of an incoming data stream and not a concept in the sense of concept detection from Sec. (**2.1**). Two major approaches exist in this area: first, the reweighing of training samples depending on a fixed or adaptive window moving over the stream [26, 50]. Second, fusion of weighted ensemble classifier from different parts of the data stream [51, 106].

2.2.2 Domain Adaptation for Concept Detection

Finally, we consider domain adaptation in concept detection, where it is commonly referred to as **Cross-domain Learning**. One of the first studies in this area was motivated by a dataset change of the TRECVID campaign. The dataset changed from the previously used *news video* (2005 + 2006) to the newly introduced *documentary video* (2007), which was leading to the situation where robust detectors for the news domain were available but could not be used naively on the new documentary domain. On the other side, the initially small amount of positive samples on the documentary dataset might not have been sufficient for new domain specific detector training.

Data Aggregation One straightforward approach to domain adaptation is *Data Aggregation*, where labeled samples from both domains are merged together before classifier training $\mathcal{D}_l = \mathcal{D}_l^s \cup \mathcal{D}_l^t$. Data aggregation – often used as a baseline [44, 114] is usually biased towards the larger dataset. However, this method can be helpful in a cold start (or kickstart) scenario [78] where an aggregated training dataset is iteratively enriched by samples from the target domain for further classifier training.

Adaptive-SVM A first domain adaptation approach, where an existing source domain classifier is directly modified, is the *Adaptive-SVM* (A-SVM) [114]. An A-SVM learns a specific *delta function* to compensate the domain change. In particular, the adapted classifier $f^a(x)$ is a combination of the source classifier $f^s(x)$ and the delta function $\Delta f(x) = \mathbf{w}^T\phi(x)$ leading to

$$f^a(x) = f^s(x) + \Delta f(x) = f^s(x) + \mathbf{w}^T\phi(x)$$

where $\mathbf{w}$ is the parameter to be derived from the labeled target domain samples $\mathcal{D}_l^t$ and $\phi(x)$ is the well-known kernel mapping function. The goal of an A-SVM is to learn a new decision boundary which is close to the original decision boundary but also separates well labeled samples on the target domain. This is reached by solving:

$$\arg\min_{\mathbf{w}} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{N}\xi_i$$

$$s.t. \quad \xi_i \geq 0, \quad y_i(f^s(x)_i + \mathbf{w}^T\phi(\mathbf{x}_i)) \geq 1 - \xi_i, \quad \forall(\mathbf{x}_i, y_i) \in \mathcal{D}^t$$

where the first term minimizes the difference between decision boundaries of the source and target domain and the second term reduces the total classification error in the target domain. This approach can also be extended to multiple source domains and may be embedded into a more general classifier adaptation framework [113]. A limitation of A-SVM, however, is its regularization constraint forcing the new decision boundary to be close to the old decision boundary learned from the source domain.

Cross-Domain SVM Another SVM-based approach has been introduced in [17, 44]. Here, a *Cross-Domain SVM* (CDSVM) aims not to learn a delta function but to directly utilize the knowledge about the source domain compressed in the support vectors $\mathcal{V}^s = \{(\mathbf{v}_1^s, y_1^s), \ldots, (\mathbf{v}_M^s, y_M^s)\}$ representing the source classifier $f^s(x)$. The underlying idea is that if a support vector $\mathbf{v}_i^s$ falls into the neighborhood of a target data sample it is likely to be drawn from a distribution similar to $\mathcal{D}^t$ and therefore help to classify $\mathcal{D}^t$. This leads to the following optimization problem:

$$\arg\min_{\mathbf{w}} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{N}\xi_i + \sum_{j=1}^{M}\sigma(\mathbf{v}_j^s, \mathcal{D}^t)\overline{\xi_j}$$

$$\begin{aligned} s.t. \quad & \xi_i \geq 0, \quad y_i(\mathbf{w}^T\phi(\mathbf{x}_i)) \geq 1 - \xi_i, \quad \forall(\mathbf{x}_i, y_i) \in \mathcal{D}^t \\ & \overline{\xi_j} \geq 0, \quad y_j^s(\mathbf{w}^T\phi(\mathbf{v}_j^s)) \geq 1 - \overline{\xi_j}, \quad \forall(\mathbf{v}_j^s, y_j^s) \in \mathcal{V}^s \end{aligned}$$

where the first term inversely is related to the margin of training examples i.e. seeks for a maximum margin between two classes, the second term measures the total classification error on the target domain and the last term computes the similarity $\sigma(\mathbf{v}_i^s, \mathcal{D}^t)$ of support vectors $\mathbf{v}_j^s$ to target domain data $\mathcal{D}^t$.

Transductive SVMs An additional approach applied in the context of concept detection are *Transductive SVMs* (TSVM) [18, 44, 113], a semi-supervised learning method. Since a large dataset on the target domain is available but only few samples are labeled it seems reasonable to employ semi-supervised learning on the following setup: $\mathcal{D}^t = \mathcal{D}^t_l \cup \mathcal{D}^t_u \quad \wedge \quad |\mathcal{D}^t_l| \ll |\mathcal{D}^t_u|$. Such an optimization problem can be formulated as solving [44]:

$$\arg\min_{\mathbf{w}} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{N_l^t} \xi_i + \overline{C}\sum_{j=1}^{N_u^t} \overline{\xi_j}$$

$$\begin{aligned} s.t. \quad & \xi_i \geq 0, \quad y_i(\mathbf{w}^T\phi(\mathbf{x}_i)) \geq 1 - \xi_i, \quad \forall(\mathbf{x}_i, y_i) \in \mathcal{D}^t_l \\ & \overline{\xi_j} \geq 0, \quad \overline{y}_j(\mathbf{w}^T\phi(\overline{\mathbf{x}}^s_j)) \geq 1 - \overline{\xi_j}, \quad \forall(\overline{\mathbf{x}}_j, \overline{y}_j) \in \mathcal{D}^t_u \end{aligned}$$

where a maximum margin constraint in the first term is balanced against the total classification error of training samples from $\mathcal{D}^t_l$ and samples $(\overline{\mathbf{x}}_j, \overline{y}_j)$ from $\mathcal{D}^t_u$ as being based on the label estimation $\overline{y_j}$.

Note, however, that one key assumption of semi-supervised learning – namely that the underlying data distribution of both domains is similar – cannot be expected to hold in concept detection. Other drawbacks are computational cost and non-convex optimization.

Domain Transfer SVM The same idea of utilizing unlabeled target domain samples inspired *Domain Transfer SVM* (DTSVM) [29]. This approach is based on simultaneously learning a kernel function and minimizing the mismatch of data distributions of source and target domain by employing the structure of the unlabeled target data $\mathcal{D}^t_u$ by Maximum Mean Discrepancy [9]. In contrast to TSVMs, this approach does not assume similar data distributions between source and target domains. However, to solve the underlying optimization problem of DTSVM, the learning algorithm uses Semi-Definite Programming (SDP), a computational expensive method which is not suitable for large datasets as they occur in web-based concept detection.

Domain Adaptation Machines A similar approach from the authors of DTSVM are *Domain Adaptation Machines* (DAMs) [28]. DAMs are multiple source learners based on Least-Squares SVM (LS-SVM) [92] and enhanced by a data-dependent smoothness regularizer leading to an adapted classifier. An advantage of DAMs are their sparse characteristic and the resulting computational efficiency. Additionally, DAMs do not depend on the assumption that source and target domain samples come from the same data distribution. However, DAMs depend on independent source classifiers to find similarities in the distribution of the unlabeled target domain. This requires the availability of such source classifiers, which leads to a great effort in providing multiple source classifiers from different domains.

Domain Adaptive Semantic Diffusion A different direction of domain adaptation in concept detection is the utilization of relations between concepts. Chang et al. proposed a *contextual model* based on source classifiers providing confidence scores which serve as input features to train a target classifier [17]. An extended version of this idea was practically used in [60], where a graph based approach called *Domain Adaptive Semantic Diffusion* (DASD) [47] was used in combination with an ontology based word similarity named *Flickr Context Similarity* (FCS) [46]. However, the aforementioned methods are particularly used for semantic context adaptation targeting the *domain-shift-of-context problem* [60] and not for direct classifier adaptation as in case of the SVM based approaches previously described. In our case, we aim to perform domain adaptation in terms of data distribution shifts and the resulting different visual appearance of the same semantic concept as illustrated in Fig. (**1.1**).

So fare the proposed domain adaptation techniques were based on SVMs and compared either domain change influences between news channels or between news vs. documentary material. In this thesis, the impact of web video as alternative source for concept detector training is investigated, a more challenging task due to its intrinsic label noise and subjectivity. Furthermore, the proposed domain adaptation approach – as being based on online learning – presents a highly efficient alternative to SVM-based methods. As a result an additional key benefit of the proposed method is its ability to incorporated user feedback into the adaptation process. Such a user feedback in combination with active learning might help to further reduce the need for labeled target domain samples.

2.3 Active Learning

In supervised learning, classifier training is performed on a labeled dataset. Prior to training such a labeled dataset must manually acquired by annotating unlabeled samples. This is an expensive and cost-intensive effort. The main goal of active learning is to select only the "most informative" samples for manual annotation and therefore to minimize the effort of labeling new datasets [77]. In particular, active learning works in iterative cycles, where each cycle consist of three steps: first, a model training. Second, a "query sample" selection based on this model, and third, the manual annotation of the selected sample. This user feedback is then included in the next cycle of active learning leading to a successively improved classifier.

2.3.1 Pool-based Methods

The first appearance of active learning alike algorithms can be tracked back to the *Query-by-Committee* (QBC) algorithm [79], where the query sample is selected according to the principle of maximal disagreement among different learners. Theoretical analysis of QBC [31] proofed a possible exponential error decrease in a learning setup where the learner observes a stream of samples and can decide for the current sample whether or not to ask for its label. A first comprehensive statistical foundation for active learning was provided by Cohn et al. [21]. Their work is focused on a statistically optimal selection of training data, motivated by minimizing the expected future classification error.

One particular family of active learning algorithms which is specifically suitable for retrieval is pool-based active learning. In pool-based active learning the learner has access to a pool of unlabeled data and can request the user to label a certain amount of instances in the pool to improve retrieval results. A straightforward method of selecting samples is *most relevant sampling* [74] as it is motivated by the idea of relevance feedback. In the context of text retrieval, Lewis and Gale [53] introduced *uncertainty sampling*, which is also known as the close-to-boundary criterion [75, 94].

Another approach coming from the text-retrieval domain is based on the estimated error reduction strategy by Roy and McCallum [73] rather than utilizing heuristic approaches like aforementioned. The error reduction strategy is comparable to Chon [21] but differs in the realization. While Cohn provided the theoretical foundation with focus on examples that can be represented in closed form and rather construct "best" samples instead of choosing from a pool, Roy and McCallum were able to solve the practical problem of efficiency by sampling estimation.

2.3.2 Applications in Image & Video Retrieval

In the context of image retrieval, Tong and Chang [15, 93] proposed a version space reduction approach for sample selection. Clustering-based approaches were presented in [61, 69], where the structure of the underlying data distribution is utilized. This provides good means for the cold start of active learning and improves performance by not labeling redundant samples belonging to the same cluster.

Active learning also finds application in video retrieval [90]. A large-scale evaluation of standard active learning sample selection methods can be found in [4], where close to ground truth detector performance could be achieved by labeling only 15% of the original TRECVID 2006 dataset. Particularly interesting is the performance improvement when taking temporal information into account for sample selection, i.e actively selecting neighborhood shots of already positive annotated shots. Ayache and Quenot also embedded active learning methods within the TRECVID collaborative annotation effort [5]. A more general view of active learning for multimedia can be found in [19, 39], where active learning is the method behind the feedback mechanism of the proposed retrieval system. Another scenario where active learning proofed to be helpful is relevance filtering of noisy web video material [11]. Here, the combination of automatic filtering techniques with successively refined labels by active learning improved concept detector performance up to expert level by refining only 25% of the positive labeled samples.

Summarizing, active learning can be employed in three practical video retrieval situations. First, in situations where only few annotations are given, active learning can help to efficiently identify and annotate the most "informative samples" to increase training data size. Second, in relevance filtering setups, where a labeled but noisy dataset is available but must be refined, and third, in combination with domain adaptation in using a robust classifier trained on a source domain to acquire training samples from a target domain.

Chapter 3

Approach

In this chapter, a novel domain adaptation approach for concept detection will be outlined. This approach is based on previous work on passive aggressive online learning [35, 66], where a similar method was introduced as a fast and accurate alternative to SVMs for concept learning. The extension of those methods to domain adaptation as proposed in this thesis is particularly designed to be highly efficient in terms of computational cost. This allows to process large amount of web video data and to incorporate domain adaptation as a part of an interactive relevance feedback loop. Such feedback on the target domain can be seen as a strong requirement in a real world domain adaptation scenario.

First, some basic notation will be introduced, followed by an overview of the proposed domain adaptation framework. After this, the underlying statistical learning procedure will be outlined and a novel adaptation method will be introduced: it will be shown how this method can be used in batch processing mode (where a set of labeled target domain samples is available) and how adaptation can be performed simultaneously with sample acquisition using active learning.

3.1 Problem Overview

For the sake of simplicity the methods throughout this chapter will be discuss in terms of a single concept. Video content is represented as a set of keyframes $x \in X$, associated with feature vectors $\mathbf{x} \in \mathbb{R}^d$. Additionally, let $(\mathbf{x}_i, y_i)$ be a sample consisting of a feature vector and a binary class label $y_i \in \{+1, -1\}$ indicating the presence of a concept. Let $\mathcal{D}^s$ denote the dataset acquired from the source domain. In our case this dataset is fully labeled i.e. $\mathcal{D}^s = \mathcal{D}^s_l$. Note that web video is defined as source domain, which is *weakly* labeled i.e. labels are only weak indicators of concept presence because of their noise and subjectivity [97]. Further, $\mathcal{D}^t$ is denoted as the target domain dataset consisting of two subsets: a labeled dataset $\mathcal{D}^t_l$ and an unlabeled one $\mathcal{D}^t_u$. Here it is assumed that $|\mathcal{D}^s_l| \gg |\mathcal{D}^t_l|$, the amount of labeled source domain samples is much higher than the amount of target domain samples. This corresponds to the practical situation, as web video is available at large scale through online portals like YouTube.

In a domain adaptation scenario, distributions of $\mathcal{D}^s$ and $\mathcal{D}^t$ are assumed to be different and therefore the classifier $f^s(\mathbf{x})$ trained on the source domain will not perform well classifying $\mathcal{D}^t_u$. However, to learn a new classifier $f^t(\mathbf{x})$ alone from the labeled samples of the target domain may not give robust performance due to the small amount of training samples. Concluding, the core question of domain adaptation is how to utilize the knowledge given by the source domain (compressed in $f^s(\mathbf{x})$) in combination with $\mathcal{D}^t_l$ to improve classification of $\mathcal{D}^t_u$.

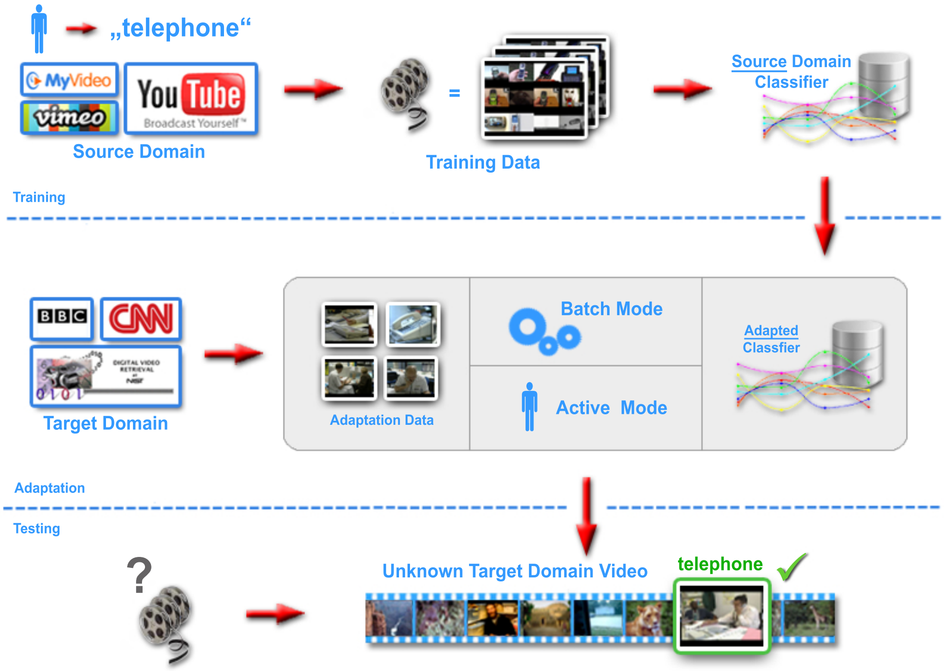


Figure 3.1: To learn a concept, the proposed system downloads videos from web video portals and trains an initial source domain classifier. This classifier is adapted efficiently either in a *batch adaptation* mode, or an *active adaptation* mode obtaining a final domain-specific detector.

3.2 Proposed Framework

The core interest of this work is to adapt web trained classifiers to specific target domains other than web video. To achieve this, the following framework is proposed as illustrated in Fig. (**3.1**). To learn a particular concept like “telephone”, training material is downloaded from an online video platform (i.e. the source domain $\mathcal{D}^s$). This material is used for initial concept detector training, resulting in a source classifier $f^s(\mathbf{x})$. The source classifier $f^s(\mathbf{x})$ is then adapted efficiently in an online fashion to a classifier $f^a(\mathbf{x})$ by utilizing a few samples from the target domain $\mathcal{D}^t$. This can be performed in two different modes:

1. *batch adaptation*, where adaptation is done on a set of a priori available labeled samples from the target domain.

2. *active adaptation*, where adaptation is alternated with sample acquisition by active learning. Such an alternated user feedback is only possible if the underlying domain adaptation approach is highly efficient. This mode aims at practical scenarios where no target domain labels are available a priori.

Note that such an adaptation is different than training an entirely new target classifier that is based purely on the labeled samples from the target domain. Here, a classifier is trained on a large training set from the source domain and adapted to a rather small set of target domain samples.

3.3 Passive Aggressive Online Learning

Because one core requirement for the illustrated domain adaptation framework is efficiency in terms of learning and classification, the passive aggressive online learning approach (PAMIR) of Grangier [35] is adapted as its underlying statistical learning method. Additionally, its online learning nature makes its particularly applicable for domain adaptation. PAMIR has also been demonstrated to serve as a competitive alternative to SVMs in concept detection [66].

3.3.1 Linear Discriminative Model

The goal of PAMIR is to find a linear projection from the feature space to a concept score. Such a projection is represented by a *weight vector* $\mathbf{w}$. Given an unlabeled sample $\mathbf{x}_i$, classification scores for a concept are calculated by a dot product:

$$f(\mathbf{x}_i) =< \mathbf{w}, \mathbf{x}_i > \tag{3.1}$$

This simple dot product schema allows a very efficient classification of unseen samples. One additional characteristic of this method is its compact representation of the model by the weight vector $\mathbf{w}$, and an efficient concept learning as illustrated next. Concept learning means here to find a suitable weight vector $\mathbf{w}$ for a concept. This is accomplished by minimizing the following optimization problem:

$$\arg\min_{\mathbf{w}} \ || \mathbf{w} ||^2 \quad + \mathcal{C} \sum_{\forall(\mathbf{x}_p,\mathbf{x}_n)\in\mathcal{D}} l(\mathbf{w}; \mathbf{x}_p, \mathbf{x}_n) \tag{3.2}$$

where $\mathbf{x}_p \in \mathcal{D}$ is a positive sample ($y_p = +1$) drawn from the training data $\mathcal{D}$ and $\mathbf{x}_n \in \mathcal{D}$ is a negative sample ($y_n = -1$), $\mathcal{C}$ being a cost parameter, and $l(\cdot)$ being the hinge loss function:

$$l(\mathbf{w}; \mathbf{x}_p, \mathbf{x}_n) = \begin{cases} 0 & \mathbf{w}(\mathbf{x}_p - \mathbf{x}_n) > 1 \\ 1 - \mathbf{w}(\mathbf{x}_p - \mathbf{x}_n) & otherwise \end{cases} \tag{3.3}$$

Intuitively, optimization of Eq. (**3.2**) results in high classification scores of videos where learned concept appears and low classification scores for videos where the concept is not present. Following, the minimization criterion directly optimizes *Average Precision* (AvgP), a standard performance metric in video concept detection.

3.3.2 Learning Algorithm

The optimization problem of finding a suitable weight vector $\mathbf{w}$ can now be solved with the help of an iterative procedure [66]:

$$\mathbf{w}^i = \arg\min_{\mathbf{w}} \ \frac{1}{2} \, || \mathbf{w} - \mathbf{w}^{i-1} ||^2 \ + \ c \; l(\mathbf{w}; \mathbf{x}_p, \mathbf{x}_n) \tag{3.4}$$

where i is the number of optimization steps, c is the hyper-parameter of the system controlling the *aggressiveness* of optimization, $\mathbf{x}_p$,$\mathbf{x}_n$ a positive and negative labeled random pair of samples and $l(\cdot)$ being the previously introduced loss function (Eq. **3.3**). In particular, Eq. (**3.4**) consist of two terms: the first term $\frac{1}{2} \, || \mathbf{w} - \mathbf{w}^{i-1} ||^2$ forces the new weight vector to be nearby to the previous one. It can be understood as a regularizer performing a smoothing between successive optimization steps. Whereas the second term $c \; l(w, (\mathbf{x}_p; \mathbf{x}_n))$ represents the ability to discriminate correctly between positive and negative samples.

Table 3.1: Outline of the learning algorithm. After an initial model setup, n iterations of model updates are performed according to randomly drawn sample pairs.

```
1. initialize weight vector with $\mathbf{w}^{i=0} = 0$
2. for $i = 1, \ldots, n$ do:
    • select randomly $(\mathbf{x}_p, \mathbf{x}_n) \in \mathcal{D}$
    • obtain new weight vector $\mathbf{w}^i$ by Eq. (3.4):
        – perform update according to Eq. (3.5) and Eq. (3.6)
```

The optimization algorithm as shown in Tab. (**3.1**) starts with iteration $i = 0$ and a weight vector initialization of $\mathbf{w}^{i=0} = 0$. At each iteration a random pair of samples $(\mathbf{x}_p, \mathbf{x}_n)$ is drawn from $\mathcal{D}$ and evaluated. For this pair the new weight vector $\mathbf{w}^i$ is obtained by solving Eq. (**3.4**). According to [35] the solution to this equation is:

$$\mathbf{w}^i = \mathbf{w}^{i-1} + \Gamma^i(\mathbf{x}_p - \mathbf{x}_n) \tag{3.5}$$

where the Lagrange multiplier Γ^i is:

$$\Gamma^i = \min\left\{\mathcal{C}, \frac{l(\mathbf{w}; \mathbf{x}_p, \mathbf{x}_n)}{|| \mathbf{x}_p - \mathbf{x}_n ||^2}\right\} \tag{3.6}$$

After a appropriate number of iterations $i = n$ the procedure is stopped and model training is finished.

3.4 Domain Adaptation

The previously introduced algorithm is an online learning method. Online learning methods build a model by processing a sequence of labeled samples, where each single sample is viewed as one instance at a time by the learner during training and therefore contributes only gradually to the learned model. This is also true for the iterative optimization of Eq. (**3.4**), where each pair of samples stands for one step of optimization. This characteristic fits smoothly into the domain adaptation framework. The idea is to first, train a source classifier by iteratively picking samples from $\mathcal{D}^s$ and then to adapt this classifier by picking samples explicitly from the target domain $\mathcal{D}^t$.

3.4.1 Batch Adaptation

In batch adaptation mode a new weight vector is learned from the target dataset $\mathcal{D}_l^t$ with the support of $\mathcal{D}^s$. A detailed description of the adaptation procedure can be found in Tab. (**3.2**): Let f^s be the classifier trained on the source domain $\mathcal{D}^s$ by optimizing Eq. (**3.2**) and substituting $\mathcal{D} = \mathcal{D}^s$. Then, adaptation of f^s is performed as a second optimization of Eq. (**3.2**), but here with $\mathcal{D} = \mathcal{D}_l^t$ and more importantly instead of initializing the weight vector with $\mathbf{w}^{s^{i=0}} = 0$, the second optimization starts with the previously learned model from the source domain: $\mathbf{w}^{a^{j=0}} = \mathbf{w}^s$. This provides a reasonable foundation for adaptation in providing knowledge from the source domain. Now, every optimization step j of Eq. (**3.4**) aims to find a weight vector $\mathbf{w}^{a^j}$ which is close to the source domain model and can separate sample pairs drawn explicitly from the target domain $\mathcal{D}_l^t$.

Table 3.2: After a first model training on the source domain (step 1. and 2.) a second optimization cycle (step 3. and 4.) performs adaptation on the target domain.

```
1. initialize source classifier training with weight vector $\mathbf{w}^{s^{i=0}} = 0$
2. for $i = 1, \dots, n$ do:
    • select randomly $(\mathbf{x}_p, \mathbf{x}_n) \in \mathcal{D}^s$
    • obtain new weight vector $\mathbf{w}^{s^i}$ by Eq. (3.4):
        – perform update according to Eq. (3.5) and Eq. (3.6)
   Once the first optimization cycle is finished the source domain
   classifier $f^s(\mathbf{x})$ is represented by the weight vector $\mathbf{w}^s$.

3. initialize target domain adaptation with weight vector $\mathbf{w}^{a^{j=0}} = \mathbf{w}^s$
4. for $j = 1, \dots, m$ do:
    • select randomly $(\mathbf{x}_p, \mathbf{x}_n) \in \mathcal{D}_l^t$
    • obtain new weight vector $\mathbf{w}^{a^j}$ by Eq. (3.4):
        – perform update according to Eq. (3.5) and Eq. (3.6)
   Once the second optimization cycle is finished the adapted classifier $f^a(\mathbf{x})$
   is represented by the weight vector $\mathbf{w}^a$.
```

Note that this approach is not simply a data aggregation approach as described in Sec. (**2.2**), where source domain data and target domain data are put together leading to one optimization procedure with $\mathcal{D} = \mathcal{D}^s \cup \mathcal{D}_l^t$. Instead, the proposed adaptation can be seen as belonging to the "function level classifier adaptation" schema proposed by Yang [113] as it seeks to learn a decision boundary close to the source domain decision boundary and tries to separate the labeled target domain samples. However, here, the computation of the new model is different due to the underlying online learning optimization so that the regularization condition of finding a "close" boundary is only valid in the scope of one single optimization iteration. Controlling the number of iterations allows the model to gradually change over time and therefore adapt to target domains which are not "close" to the source domain.

3.4.2 Adaptation Stopping Criterion

One remaining question is: how much adaptation is enough i.e. how many iterations of optimization steps should be performed? While it can be expected that adaptation improves concept detector performance in general, it may have negative impact on some concepts where the source domain may already provide comprehensive training material resulting in a robust source classifier. In such cases an adaptation could harm the final classification performance, an effect which is known as "negative transfer" [72] in transfer learning.

Domain adaptation according to the proposed method consists of two major optimization cycles, namely training on the source domain and adaptation to the target domain. For source domain training, the number of iterations is fixed. For adaptation, however, the question of when to stop is tackled by testing the following rules:

1. **oracle**, where adapted classifier $f^a(\mathbf{x})$ performance is evaluated on the target domain $\mathcal{D}^t_u$. Adaptation is stopped in the very moment when performance on the target domain is best. While this is not feasible in reality it provides an upper bound for adaptation performance.

2. **sample**, where a small set of additional labels from the target domain data $\mathcal{D}^t_u$ is made available for adaptation evaluation. This criterion is similar to oracle but uses a much smaller fraction of $\mathcal{D}^t_u$ ($\approx 5\%$), making it feasible in a real application scenario.

3. **predict**, where a validation set from $\mathcal{D}^t_l$ is defined on which adaptation performance can be evaluated and accordingly stopped if degradation is observable.

4. **fix**, where a fixed number of iterations is set. This straightforward setup always assumes that domain adaptation on the target domain is necessary and that this should be performed with a fixed number of iterations. A good value was empirically evaluated in [66] showing a stable behavior of the online learning optimization. Therefore, the number of iterations is set s to the same value, namely $i = 10^6$ for both source domain training and target domain adaptation.

3.5 Domain Adaptation extended by Active Learning

While so far labeled samples from the target domain $\mathcal{D}^t_l$ are assumed to be available, in a real world application this may not be the case. In such a situation, label acquisition from the target domain $\mathcal{D}^t$ is necessary but should be kept to a minimum. Taking also into account that not all samples from the target domain are equally informative, an efficient sample selection is preferable. For this purpose active learning is particularly suitable, where samples are selected according to different heuristics.

3.5.1 Relevance Feedback as a Wrapper

In the following setup, a manual labeling of selected samples from $\mathcal{D}^t_u$ is placed as a wrapper around the proposed domain adaptation procedure. Note that initially $\mathcal{D}^t_l = \emptyset$ i.e. we do not have any labeled samples from the target domain.

The procedure is illustrated in detail in Tab. (**3.3**): initially a source classifier $f^s(\mathbf{x})$ is trained on the source domain $\mathcal{D}^s$. This classifier provides an initial set of classifications scores $p_i^{j=1}$ on the target domain $\mathcal{D}^t_u$. For each iteration j of adaptation a sample from $\mathcal{D}^t_u$ is selected according to an active learning criterion based on the previous scores p_i^j. This sample is manually labeled and belongs now to $\mathcal{D}^t_l$ being available for the next batch of domain adaptation resulting in an adapted classifier $f^{a^j}(\mathbf{x})$. This new classifier will provide new scores p_i^{j+1} for the next iteration of active learning sample selection. Continuing further, this procedure increases the amount of labeled samples from the target domain successively, providing an improved basis for adaptation.

3.5.2 Sample Selection Methods

In the literature, different active learning strategies for sample selection exist. Here, we compare the ones that proofed to be successful in previous concept detection experiments [4]:

Table 3.3: Active learning wrapped around domain adaptation selects informative samples from the target domain for manual annotation. Once the sample is labeled, the system takes this newly acquired knowledge into account for the next iteration of adaptation.

```
1. train source classifier $f^s(\mathbf{x})$ on source domain $\mathcal{D}^s$
2. $\forall \mathbf{x}_i \in \mathcal{D}^t_u$ obtain classification scores $p_i^{j=1}$ using $f^s(\mathbf{x}_i)$
3. for $j = 1, \ldots, m$ do:
    • select sample $s^*$ according to an active learning criterion $Q$:
          $s^* := \arg\max_i \quad Q(p_i^j)$
    • get label $y_{s^*}$
    • add sample $s^* = (\mathbf{x}_{s^*}, y_{s^*})$ to $\mathcal{D}^t_l$
    • perform domain adaptation on $\mathcal{D}^t_l$ obtaining an adapted
      classifier $f^{a^j}(\mathbf{x})$
    • $\forall \mathbf{x}_i \in \mathcal{D}^t_u$ obtain classification scores $p_i^{j+1}$ using $f^{a^j}(\mathbf{x}_i)$
```

1. **random sampling**: samples are selected randomly.
2. **most relevant**: samples are selected which are most likely to be relevant and are therefore associated with the highest posterior [74]:

$$Q_{REL}(p_i^j) := p_i^j$$

3. **uncertainty**: samples are selected for which the relevance filtering method is least confident, i.e. $p_i^j \approx 0.5$ [53]:

$$Q_{UNC}(p_i^j) := 1 - |p_i^j - 0.5|$$

Note that the proposed active learning approach does not incorporate shot neighborhood information into the sample selection mechanism as in [4].

Chapter 4

Experimental Results

In the previous section, a domain adaptation framework for concept detection has been proposed to tackle the domain change problem. This approach is studied in the following experiments, whereas the focus is on web video downloaded from YouTube as a source domain and two datasets of television content from the TRECVID campaign [63] as the target domains. In particular, this raises three key questions: first, how strong is the impact of the domain change in terms of performance degradation, second, how successful is domain adaptation in terms of detector performance and efficiency and third, how much can active learning help when adaptation data must be acquired manually.

4.1 Datasets

Several datasets are used to benchmark domain adaptation for concept detection (an overview is given in Tab. (**4.1**)). Two datasets belong to the source domain: *Web Video*, and two datasets to each of the two target domains: the TRECVID 2007 *Sound & Vision (S&V)* data and the TRECVID 2005 *Linguistic Data Consortium News (LDC News)* data. Datasets are additionally denoted as "training data", where the source classifier is trained on, others as "adaptation data", where labeled samples from the target domain are given or "test data" serving for performance evaluation. As a concept vocabulary, the 20 test concepts from the TRECVID 2009 High-Level Feature Extraction task[1] are used as outlined in App. (**A**). In the following the datasets are described in more detail.

4.1.1 Web Video

Web video material was retrieved from YouTube through the provided API[2]. As already mentioned in Sec. (**1.2**), data acquisition from YouTube requires a mapping of concept definitions to keywords used in YouTube queries. Representative keywords and YouTube categories were chosen manually for each concept and are used for video retrieval from YouTube. For example, to retrieve video clips for the concept "bus", the keyword "bus" excluding the keywords "van", "suv", "vw" and "ride" were used. Additionally, this query was narrowed down to the YouTube category "autos & vehicles" (for a detailed overview, please refer to App. (**B**)). For each concept, 150 video clips were downloaded, obtaining the dataset `yt-direct` with 120 hours total length and 2500 videos (for some concepts not all requested 150 videos could be downloaded from YouTube). A simple change detection approach for keyframe extraction resulted in about $50,000$ keyframes.

[1] http://www-nlpir.nist.gov/projects/tv2009/tv9.hlf.for.eval.txt

[2] http://code.google.com/apis/youtube/overview.html

Table 4.1: Overview of datasets used in the experiments.

	Source Domain:		**Target Domain:**	
	raw	refined	S&V	LDC News
training data ($\mathcal{D}^s$)	`yt-direct`	`yt-refined`		
adapt. data ($\mathcal{D}^t_l$)			`tv7-devel`	`tv5-lscom-devel`
test data ($\mathcal{D}^t_u$)			`tv7-test`	`tv5-lscom-test`

As the focus is on web video, domain change is also studied in combination with *label noise*, given the fact that web-based training sets contain significant amounts of non-relevant material which is another challenge for web-based concept detection. To evaluate if domain change or label noise has a stronger impact on system performance, a second manually refined dataset for the following 12 concepts was acquired: "airplane flying","boat ship", "bus", "cityscape", "classroom", "demonstration", "doorway", "female human face closeup", "people dancing", "person eating", "person playing soccer", "traffic intersection". For these concepts, only keyframes were kept that were manually validated to show the concept as defined by TRECVID. The resulting dataset of about 29,000 keyframes (`yt-refined`) allows to study concept detection with domain change but free of label noise.

4.1.2 TRECVID Sound & Vision (S&V)

As a first target domain dataset the TRECVID S&V dataset (`tv7`) is used. This dataset consists of documentary video and was initially used in the TRECVID 2007 evaluation. It contains 50 hours of training data (`tv7-devel`) and 50 hours of test data (`tv7-test`). As in TRECVID the basic unit of evaluation is a shot, the dataset was segmented into the provided ground truth shot sequences [67] and multiple keyframe per shots were extracted, leading to about 68,000 keyframes. Further, the publicly available annotation set from the TRECVID collaborative effort [5] was used as label information. Comparing this expert labeled dataset to `yt-direct`, annotation quality is much higher than for raw web video, but many concepts are very rare i.e. there are ten time more negative samples than positive ones [16]. Correspondingly, the majority of the keyframes does not show any target concept. This renders domain adaptation challenging, considering positive labels as the core information from the target domain.

4.1.3 TRECVID Language Data Consortium News (LDC News)

The second target domain dataset is the TRECVID LDC News dataset (`tv5`) as introduced in the TRECVID 2005 campaign. This dataset contains news broadcasts from 6 different TV channels. Annotations were downloaded from the LSCOM website [56] for the 86 hours of the development set, which was than splitted into two equally sized datasets: `tv5-lscom-devel` and `tv5-lscom-test` containing each about 36,000 keyframes. The splitting was done chronological i.e. `tv5-lscom-devel` was broadcasted before `tv5-lscom-test`. Because the concept vocabulary of TRECVID 2009 is used in the experiments, this datasets consist of the following 12 annotated concepts: "airplane flying","boat ship", "bus", "cityscape", "classroom", "demonstration", "hand", "infant", "nighttime", "person playing soccer", "singing", "telephone". Comparing this very specific dataset to `yt-direct`, the same holds as for the `tv7` dataset: a high annotation quality and a sparse positive label distribution throughout the `tv5` data. Further, it is expected that the typical visual appearance of news broadcast (e.g the letterbox at the bottom, anchorman and recurrent TV channel logos) will not be covered in the YouTube datasets, rendering adaptation to this dataset even more challenging.

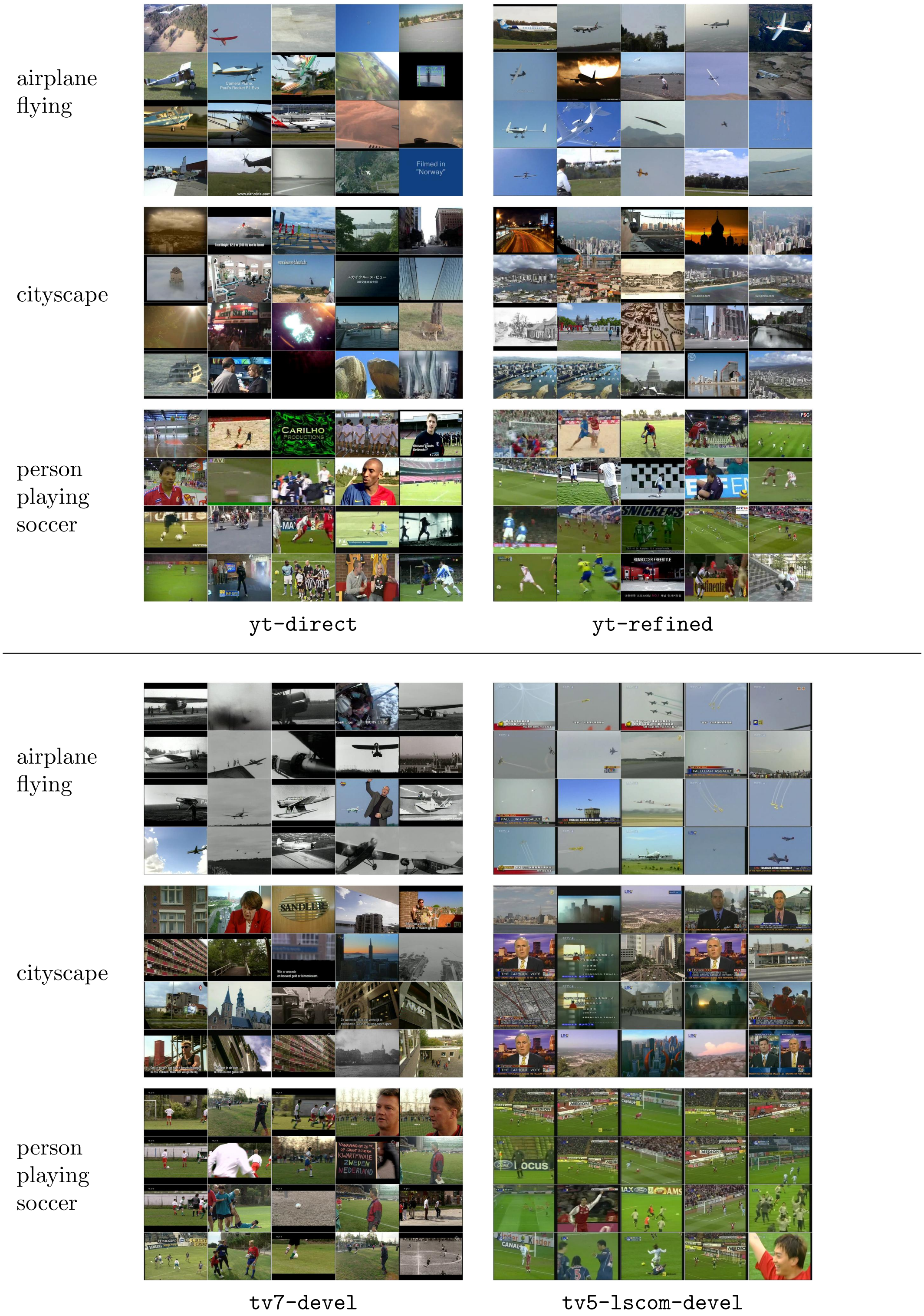


Figure 4.1: Sample keyframe from the different datasets for the concepts: "airplane flying", "cityscape" and "person playing soccer". The top part illustrates the label noise problem when comparing `yt-direct` and `yt-refined` with each other. It can be seen that video material downloaded from YouTube contains significant amounts of non-relevant content. The bottom part illustrates the target domain datasets. While the `tv7` dataset consist of documentary content (e.g. black & white video clips), the `tv5` datasets has the typical visual appearance of news broadcast with its letterbox graphics at the bottom of the screen.

The visual appearance of the different datasets is shown in Fig. (**4.1**). For each dataset a random selection of keyframes for the concept: "airplane flying", "cityscape" and "person playing soccer" is given. As displayed, the raw YouTube dataset (`yt-direct`) contains many noisy keyframes which are non-relevant, while the manually refined YouTube dataset (`yt-refined`) is providing higher annotation quality. Further, the `tv7` and `tv5` datasets illustrate significant domain changes as seen by their different visual appearance compared to `yt-direct` and `yt-refined`. While the `tv7` dataset contains typical documentary material i.e. much black & white content, the `tv5` dataset shows the expected news broadcasting look i.e. a letterbox graphics at the bottom.

4.1.4 Feature Representation

In all experiments we employ the well-known bag-of-visual-words approach for feature extraction [80, 102]: for each keyframe a regular patch sampling was performed at several scales, describing each patch by SIFT [55]. Patches were matched against a 2000-dimensional visual word codebook – build by K-Means – forming a feature histogram representing the keyframe. Further, as all classification is based on keyframes but performance evaluation is done on shot level, an averaging of keyframe scores to video shot scores was performed.

4.2 Domain Change Impact

In a first experiment we quantify the impact of domain changes by comparing classification performance of systems beings trained on the source domain vs. systems being trained on the target domain.

4.2.1 Setup

Two runs are performed for each target domain: (1) a classifier trained on `yt-direct` and (2) a classifier trained on `tv7-devel` (or `tv5-lascom-devel`). Both classifiers are tested against their corresponding target domains: `tv7-test` (or `tv5-lscom-test`). For each setup, two statistical models are evaluated: the PAMIR method, as outlined in Sec. 3.3 with $i = 10^6$ as the fixed number of optimization steps and $\mathcal{C} = 0.001$, and SVMs [76], with a χ^2 kernel ($\mathcal{C}$ and γ are evaluated in a grid-search cross validation). SVM scores were mapped to class posterior estimates using LIBSVM [14].

4.2.2 Results

As a performance measure, average precision is used, i.e. the area under the recall-precision curve over the ranked list of test videos. By averaging over all 20 concepts, the mean average precision (MAP) is obtained, a standard evaluation metric for video concept detectors [64]. Quantitative results for the `tv7` and `tv5` datasets are given in Fig. (**4.2**), as averaged over 5 repeated runs. It can be seen for the `tv7` data, that the domain change leads to a significant performance drop in MAP by up to 6.1% for PAMIR (which corresponds to a relative loss of 60%) and up to 6.75% for SVMs (relative loss 55%) when training and test data are not from the same domain. A similar behavior is observable for the `tv5` data, leading to a performance drop in MAP by up to 4.4% for PAMIR (which corresponds to a relative loss of 80%) and up to 7.5% for SVMs (relative loss of 85%).

Considering each single concept (Tab. (**4.2**)), similar performance degradations can be observed. However, four exceptions have to be pointed out when performing adaptation to the `tv7` target domain: "demonstration or protest", "female human face closeup", "person playing a musical instrument" and "person playing soccer". For these concepts the training on `yt-direct` leads to a more robust classifier than training on `tv7-devel`.

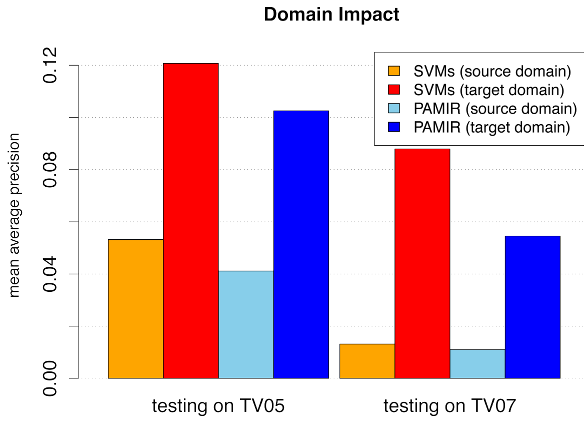


Figure 4.2: Results illustrating the domain change impact for SVMs and PAMIR based concept detectors. A significant performance drop can be observed when training and test data are not from the same domain.

Table 4.2: Results (AvgP) of domain impact for the tv7 and tv5 datasets.

	tv7				tv5			
Concept	**Source-SVM**	**Source-PAMIR**	**Target-SVM**	**Target-PAMIR**	**Source-SVM**	**Source-PAMIR**	**Target-SVM**	**Target-PAMIR**
airplane flying	0.05	0.035	0.042	**0.113**	0.005	0.010	**0.166**	0.075
boat ship	0.061	0.048	**0.188**	0.127	0.005	0.005	**0.057**	0.017
bus	0.028	0.012	**0.036**	0.014	0.005	0.005	**0.008**	0.005
chair	0.008	0.007	0.010	**0.011**	-	-	-	-
cityscape	0.089	0.079	**0.237**	0.212	0.006	0.005	**0.042**	0.023
classroom	0.005	0.007	0.016	**0.013**	0.004	0.006	0.130	**0.172**
demo. or protest	**0.110**	0.083	0.053	0.012	0.013	0.022	0.025	**0.029**
doorway	0.003	0.004	**0.030**	0.016	-	-	-	-
female human face closeup	**0.041**	0.023	0.029	0.016	-	-	-	-
hand	0.063	0.047	0.144	**0.146**	0.003	0.003	**0.048**	0.023
infant	0.031	0.037	**0.053**	0.009	0.000	0.001	0.001	**0.002**
nighttime	0.077	0.089	**0.145**	0.140	0.109	0.065	**0.280**	0.174
people dancing	0.006	0.006	**0.008**	0.007	-	-	-	-
person eating	0.004	0.004	**0.445**	0.425	-	-	-	-
person playing a musical instrument	**0.052**	0.032	0.022	0.037	-	-	-	-
person playing soccer	**0.387**	0.261	0.211	0.071	0.003	0.005	**0.150**	0.099
person riding a bicycle	0.009	0.014	0.253	**0.313**	-	-	-	-
singing	0.017	0.019	**0.029**	0.024	0.002	0.002	**0.092**	0.035
telephone	0.002	0.002	0.016	**0.018**	0.004	0.002	**0.055**	0.003
traffic intersection	0.021	0.014	**0.447**	0.327	-	-	-	-
MAP	0.053	0.041	**0.121**	0.103	0.013	0.011	**0.088**	0.055

4.3 Domain Adaptation

As shown in the previous experiments, performance of web-based concept detectors degrades significantly when they are applied to target domains different than web video. The purpose of this experiment is to study domain adaptation in a controlled setting where the size $\alpha \in \{0.0, \ldots, 1.0\}$ of adaptation data ($\mathcal{D}_l^t$) is predefined with respect to the number of positive samples per concept. Here, $\alpha = 0.0$ corresponds to applying the source domain classifier $f^s(\mathbf{x})$ with no adaptation and $\alpha = 1.0$ to perform domain adaptation on the entire adaptation data $\mathcal{D}_l^t$. For any number in between, the total amount of both positive and negative adaptation samples is reduced to a fraction of α. In particular, small fractions of α are interesting for adaptation because such a situation reflects real application scenarios where only few samples from the target domain are available and adaptation of robust source classifier is highly desired.

4.3.1 Setup

Similar to the previous experiments, 5 runs are executed for each of the methods described next. For this, five different randomly compiled datasets (one for each run) are generated for $\alpha \in \{0.1, \ldots, 0.9\}$. For $\alpha = 1.0$ the full adaptation data is used in each of the 5 runs. Domain adaptation experiments are evaluated for each of the methods described below:

- **Target-SVM**: the control run setup from the fist experiment, performed on different α values. For this control run, $\alpha = 1.0$ is equivalent to target domain classifier training, whereas $\alpha = 0.0$ is equivalent to random guessing.
- **Target-PAMIR**: same as **Target-SVM** but for the PAMIR method.
- **Aggregation-SVM**: a simple data aggregation i.e. training a classifier on both the source domain and the target domain (i.e., $\mathcal{D}^s \cup \mathcal{D}_l^t$).
- **Aggregation-PAMIR**: same as **Aggregation-SVM** but for the PAMIR method.
- **Adapt-SVM**: an SVM based domain adaptation approach by Yang [114]. An RBF kernel was used and an optimal $\mathcal{C}$ was evaluated using cross validation[3].
- **Adapt-PAMIR**: the domain adaptation extension based on PAMIR from Sec. (**3.4**) with $\mathcal{C} = 0.001$ and $i = 10^6$ iterations for source classifier training, whereas the number of adaptation steps was determined by the proposed stopping rule from Sec. (**3.4**).

4.3.2 Results

Adaptation is evaluated against two different target domains: S&V (`tv7`) and LDC News (`tv5`). Results are given in Fig. (**4.3**) and Tab. (**4.3**) for `tv7` and in Fig. (**4.5**) and Tab. (**4.5**) for `tv5`. Concept detection accuracy on the target domain $\mathcal{D}_u^t$ is plotted against α. For *Adapt-PAMIR*, two stopping rules are tested, namely the *oracle* runs and the *fix* runs as defined in Sec. (**3.4**). These two runs define the upper and lower bound of the proposed domain adaptation approach. As adaptation performance depends on the adapted concept, detailed results on concept level are also given for a full adaptation ($\alpha = 1.0$). For `tv7` the plots can be found in Fig. (**4.4**) and Tab. (**4.4**) and for `tv5`, they can be found in Fig. (**4.6**) and Tab. (**4.6**). Detailed adaptation results on concept level for the remaining $\alpha = \{0.1 \ldots 0.9\}$ can be found in App. (**C**). Furthermore, concept detector performance before and after adaptation is illustrated and the effect of early stopping is measured. This is of particular interest because of the performance gains observed in Sec. (**4.2**) for some concepts when trained exclusively on the source domain.

[3] The author would like to thank Jun Yang for his help and tips regarding Adaptive-SVMs

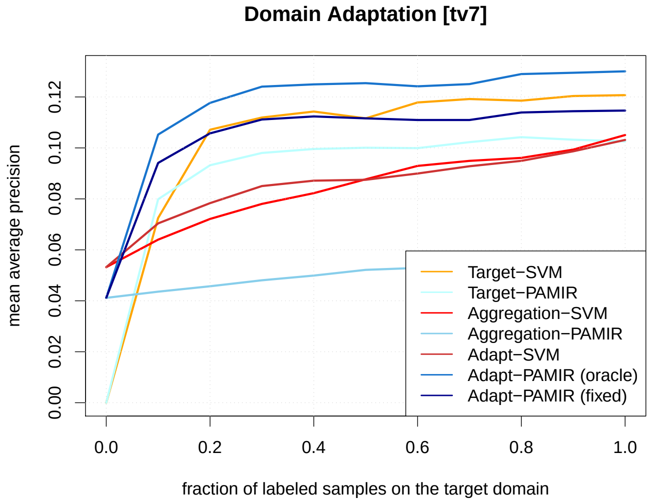


Figure 4.3: Results of domain adaptation for `tv7` data. Performance is plotted against the size α of the adaptation dataset, where *Adapt-PAMIR* (oracle) outperforms all other methods.

Table 4.3: Results in MAP of domain adaptation runs ($n = 34,000$) on `tv7` averaged over all concepts for all values of α.

Fraction α	**0.0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1.0**
Target-SVM	0.005	0.072	0.107	0.112	0.114	0.112	0.118	0.119	0.119	0.120	0.121
Target-PAMIR	0.005	0.080	0.093	0.098	0.099	0.100	0.100	0.102	0.104	0.103	0.102
Aggr-SVM	**0.053**	0.064	0.072	0.078	0.082	0.088	0.093	0.095	0.096	0.099	0.105
Aggr-PAMIR	0.041	0.044	0.046	0.048	0.050	0.052	0.053	0.054	0.056	0.057	0.056
Adapt-SVM	**0.053**	0.070	0.078	0.085	0.087	0.087	0.090	0.093	0.095	0.099	0.103
Adapt-PAMIR*	0.041	**0.105**	**0.118**	**0.124**	**0.125**	**0.125**	**0.124**	**0.125**	**0.129**	**0.129**	**0.130**
Adapt-PAMIR$^+$	0.041	0.094	0.106	0.111	0.112	0.112	0.111	0.111	0.114	0.114	0.115

**oracle*, $^+$*fixed*

A first observation for `tv7` is that the SVM classifier (*Target-SVM* run) performs uniformly better when trained on the target domain than the linear but more efficient PAMIR classifier (*Target-PAMIR*). This already has been observed in [66]. The next observation is that data aggregation (*Aggregation-SVM* and *Aggregation-PAMIR* runs) does not truly overcome the domain change problem. Finally, when considering domain adaptation, the *Adapt-PAMIR* run outperforms the *Target-PAMIR* run, where a stable performance increase of 2.8% (a relative gain of 20%) could be measured for all values of α. Comparing no adaptation and full adaptation, the *Adapt-PAMIR* run shows an improvement from 4.1% to 13.0% (oracle run), which corresponds to a relative gain of 325%. The more surprising result is, however, that the *Adapt-PAMIR* run outperforms the *Target-SVM* run by 3.2% when starting adaptation ($\alpha = 0.1$) and by 1.0% when finishing adaptation. Concluding, the kickstart setup where a web-based classifier is used for adaptation is demonstrated to perform better than training only on the target domain, even if using a stronger classifier e.g. an SVM.

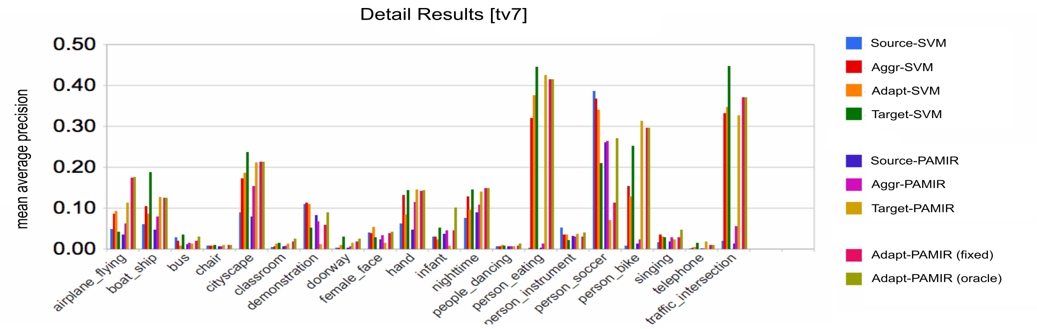


Figure 4.4: Results (AvgP) of domain adaptation on `tv7` per concept for $\alpha = 1.0$.

Table 4.4: Results showing AvgP per concept of domain adaptation runs for $\alpha = 1.0$ on `tv7`. Note that for comparison reasons, results for the source domain runs can be found in Tab. (**4.2**).

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR-fix**	**Adapt-PAMIR-oracle**
airplane flying	0.042	0.113	0.086	0.062	0.093	0.175	**0.176**
boat ship	**0.188**	0.127	0.105	0.080	0.086	0.125	0.126
bus	**0.036**	0.014	0.021	0.015	0.008	0.021	0.030
chair	0.010	0.011	0.007	0.006	0.008	0.012	**0.012**
cityscape	**0.237**	0.212	0.173	0.154	0.187	0.213	0.213
classroom	0.016	0.013	0.006	0.008	0.013	0.018	**0.026**
demo. or protest	0.053	0.012	**0.113**	0.067	0.109	0.060	0.089
doorway	**0.030**	0.016	0.004	0.007	0.010	0.018	0.025
female human face closeup	0.029	0.016	0.039	0.034	**0.055**	0.039	0.042
hand	0.144	**0.146**	0.133	0.115	0.085	0.143	0.144
infant	0.053	0.009	0.029	0.046	0.023	0.046	**0.102**
nighttime	0.145	0.140	0.129	0.109	0.097	**0.149**	**0.149**
people dancing	0.008	0.007	0.006	0.007	0.011	0.008	**0.014**
person eating	**0.445**	0.425	0.320	0.013	0.377	0.415	0.415
person playing a musical instrument	**0.052**	0.037	0.035	0.031	0.036	0.031	0.041
person playing soccer	**0.387**	0.071	**0.368**	0.264	0.341	0.114	0.272
person riding a bicycle	0.253	**0.313**	0.154	0.023	0.128	0.297	0.297
singing	0.029	0.024	0.035	0.028	0.031	0.028	**0.047**
telephone	0.016	**0.018**	0.003	0.002	0.005	0.011	0.011
traffic intersection	**0.447**	0.327	0.332	0.056	0.348	0.372	0.372
MAP	0.121	0.103	0.105	0.056	0.103	0.115	**0.130**

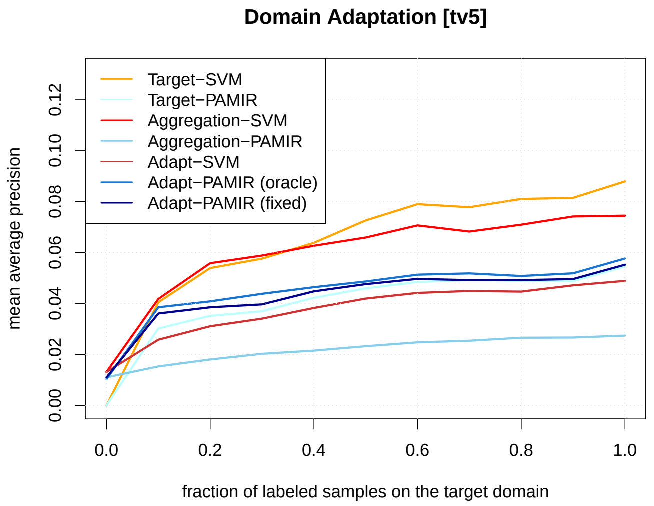


Figure 4.5: Results of domain adaptation on `tv5` data. Performance is plotted against the size α of the adaptation dataset. Domain adaptation with *Adapt-PAMIR* outperforms *Target-PAMIR* but not the *Target-SVM* and *Aggr-SVM* runs.

Table 4.5: Results in MAP of domain adaptation runs ($n = 37,000$) on `tv5` averaged over all concepts for all values of α.

Fraction α	**0.0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1.0**
Target-SVM	0.004	0.041	0.054	0.058	**0.064**	**0.073**	**0.079**	**0.078**	**0.081**	**0.081**	**0.088**
Target-PAMIR	0.004	0.030	0.035	0.037	0.042	0.046	0.048	0.049	0.049	0.049	0.055
Aggr-SVM	**0.013**	**0.042**	**0.056**	**0.059**	0.063	0.066	0.071	0.068	0.071	0.074	0.075
Aggr-PAMIR	0.011	0.015	0.018	0.020	0.022	0.023	0.025	0.025	0.027	0.027	0.027
Adapt-SVM	**0.013**	0.026	0.031	0.034	0.038	0.042	0.044	0.045	0.045	0.047	0.049
Adapt-PAMIR*	0.011	0.039	0.041	0.044	0.046	0.049	0.051	0.052	0.051	0.052	0.058
Adapt-PAMIR$^+$	0.011	0.036	0.039	0.040	0.045	0.048	0.050	0.049	0.049	0.050	0.055

$^*oracle$, ^+fixed

Considering `tv5` data, the proposed domain adaptation method (*Adapt-PAMIR*) adapts to the target domain, outperforming uniformly the target domain based PAMIR classifier (*Target-PAMIR*) for all values of α. This results in an improvement over no adaptation with 1.1% as trained on the source domain to 5.8% after full adaptation using the oracle stopping rule, which corresponds to a relative performance gain of 400%. However, the proposed adaptation method was not able to outperform the target domain based SVM classifier (*Target-SVM*), which delivers a MAP 3% higher than the *Adapt-PAMIR* run. A further observation is that early stopping is not as important as on the `tv7` data. A reason for this may be that the source domain is not able to capture the visual appearance of the `tv5` domain. This can also be observed on concept level in Tab. (**4.6**) where adaptation (*Adapt-PAMIR*) only for the concept “classroom” outperforms a target domain trained SVM (*Target-SVM*).

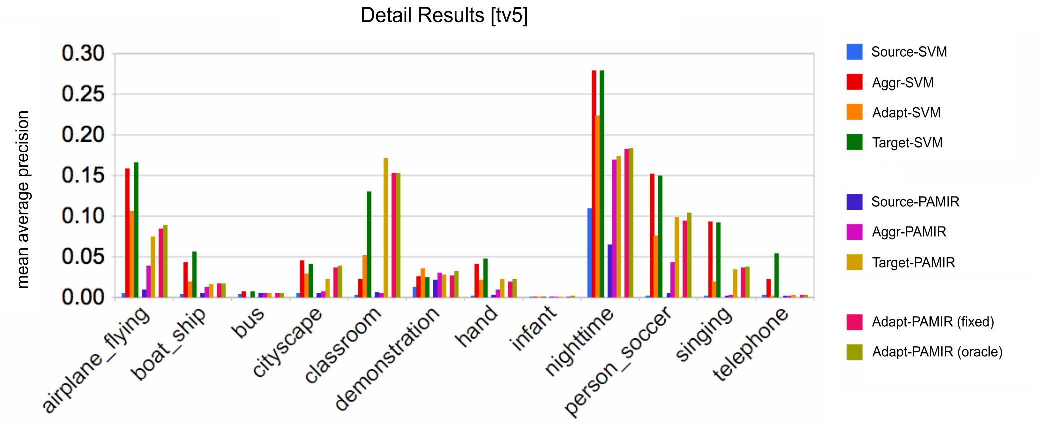


Figure 4.6: Results (AvgP) of domain adaptation on `tv5` data per concept for $\alpha = 1.0$.

Table 4.6: Results showing AvgP per concept of domain adaptation runs for $\alpha = 1.0$ on `tv5`. Note that for comparison reasons, results for the source domain runs can be found in Tab. (**4.2**).

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR-fix**	**Adapt-PAMIR-oracle**
airplane flying	**0.167**	0.075	0.159	0.040	0.106	0.085	0.090
boat ship	**0.057**	0.017	0.043	0.013	0.019	0.017	0.017
bus	**0.008**	0.005	**0.008**	0.005	0.000	0.005	0.006
cityscape	0.042	0.023	**0.045**	0.007	0.030	0.037	0.039
classroom	0.130	**0.172**	0.023	0.006	0.053	0.153	0.154
demo. or protest	0.025	0.028	0.026	0.030	**0.036**	0.027	0.033
hand	**0.048**	0.023	0.042	0.010	0.021	0.020	0.023
infant	0.001	**0.002**	0.001	0.001	0.001	0.001	**0.002**
nighttime	**0.280**	0.174	0.279	0.169	0.224	0.183	0.183
person playing soccer	0.150	0.099	**0.152**	0.043	0.076	0.095	0.105
singing	0.092	0.035	**0.093**	0.003	0.020	0.036	0.038
telephone	**0.055**	0.003	0.023	0.002	0.002	0.003	0.003
MAP	**0.088**	0.055	0.075	0.027	0.049	0.055	0.058

A visualization of the adaptation performance by *Adapt-PAMIR* for $\alpha = 1.0$ can be seen in Fig (**4.7**) for `tv7` (top) and for `tv5` (bottom). The image mosaics illustrate top ranked keyframes provided by the source classifier (top row) and top ranked keyframes provided by the adapted classifier (bottom row). As seen for the `tv7` data, domain adaptation leads to an improved classifier for the concepts “cityscape”, “airplane flying” whereas for the concept “person eating” domain adaptation leads to an adaptation to the redundant material in the dataset which is not desired but indeed provides better classification performance. For the `tv5` data, a similar behavior could be observed. Domain adaptation for the concepts “cityscape”, “airplane flying” and “classroom” show improved top ranked keyframes. Par-

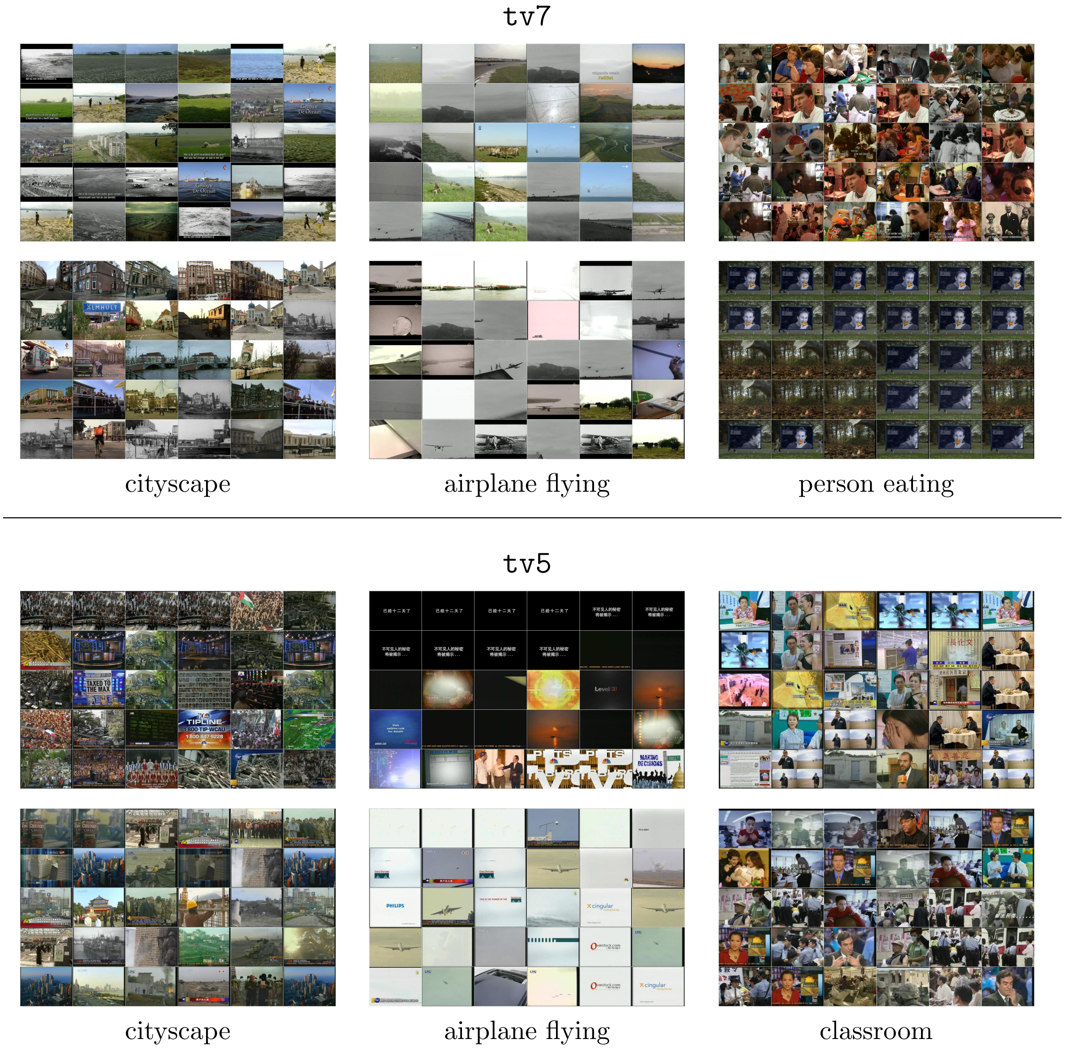
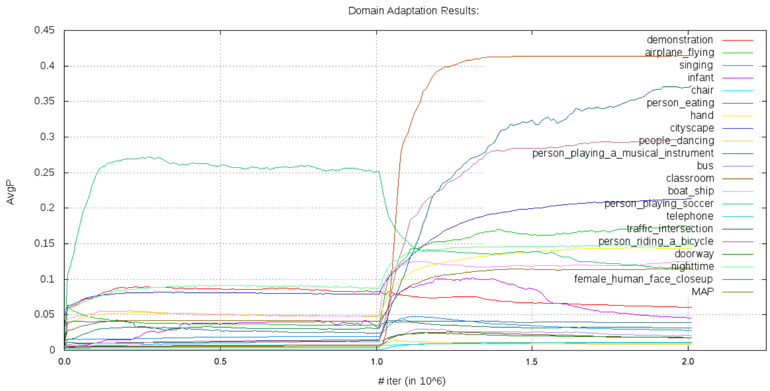


Figure 4.7: Visualization of source domain classifier results (top) vs. adapted classifier results (bottom). For some concept the visual appearance change successfully during adaptation. For other concepts, the classifiers adapted to redundant material in the dataset (right).

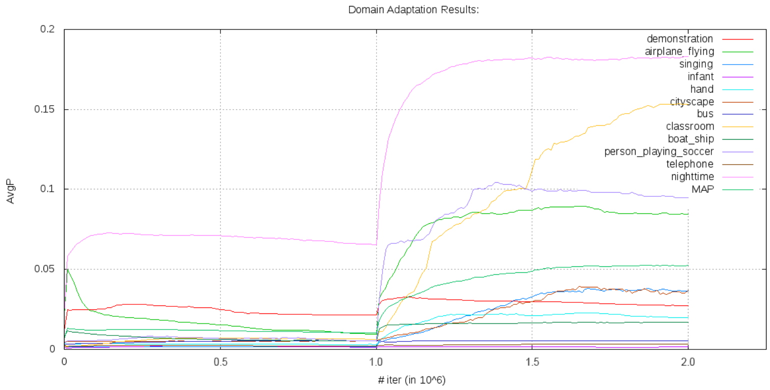
ticular for the concepts "cityscape" and "airplane flying" the web-based classifier is based on wrong clues about the actual visual appearance of the concept in the target domain.

4.3.3 Early Stopping

Evaluating the proposed early stopping rules for domain adaptation (*Adapt-PAMIR*), the following results were obtained. In general, a successful early stopping is important because it prevents negative transfer as observed in the domain impact experiment on `tv7` for the concepts "demonstation or protest", "female human face closeup", "person playing a musical instrument" and "person playing soccer". In Fig. (**4.8**) a visualization of MAP development over all target concepts is plotted against the number of iterations i for `tv7` (top) and `tv5` (bottom). Here, the first $i = 10^6$ iterations train the source classifier and the next $i = 10^6$ iterations are dedicated to domain adaptation on the full adaptation data ($\alpha = 1.0$).



Performance (tv7)	
oracle	13.01
sample	12.45
predict	11.33
fix	11.47
no adapt.	4.12



Performance (tv5)	
oracle	5.77
sample	5.70
predict	5.47
fix	5.53
no adapt.	1.11

Figure 4.8: Results of domain adaptation as performed by *Adapt-PAMIR* for a $\alpha = 1.0$ on the `tv7` data (top) and the `tv5` data (bottom). AvgP is plotted against the number of iterations. While for a majority of concepts performance increases during adaptation, for some a negative transfer in form of performance loss can be observed. To prevent this, different early stopping rules have been proposed leading to the MAP values illustrated in the right tables.

Particularly, right after beginning adaptation a strong improvement can be observed for several concepts of `tv7`, whereas for some concepts adaptation leads to the known performance loss on this domain. A good stopping rule should identify such negative transfer and stop adaptation for such concepts. Results of the proposed stopping rules and a full adaptation ($\alpha = 1.0$) can be seen in the tables of Fig. (**4.8**). A similar behavior can be observed for `tv5` i.e a strong performance improvement after starting domain adaptation. However, here, less negative transfer is visible which leads to a rather small margin between *oracle* runs and *fixed* runs. Concluding, for `tv5` early stopping is less important than for the `tv7` data. This can be interpreted as training on a weak source domain compared to the target domain (as observed for `yt-direct` vs. `tv5`) i.e. the source domain is not providing much new knowledge for concept learning.

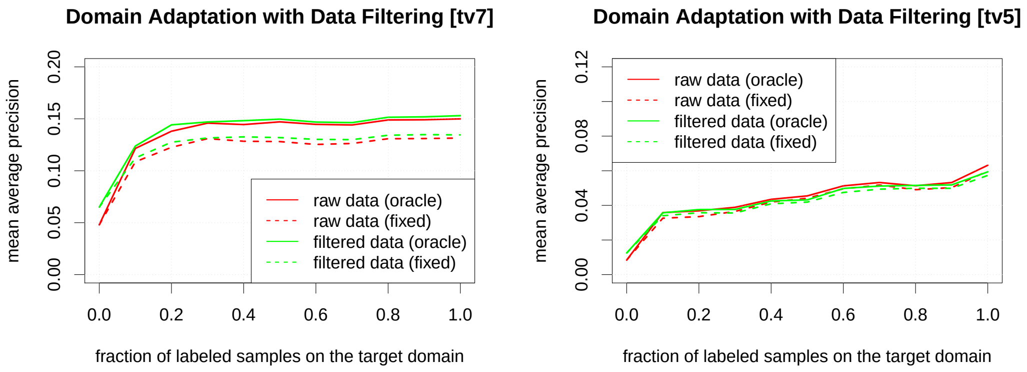


Figure 4.9: Comparing domain adaptation performance of systems initially trained on noisy web video (red) and on refined web video (green) for `tv7` (left) and `tv5` (right). Noisy training material has initially a negative influence on system performance, which, however, is negligible after domain adaptation

4.4 Label Noise Problem vs. Domain Change Problem

In the previous experiments, classifier trained on web video as source domain were adapted to specific target domains. In the following, we address another aspect, namely that web video is known to be noisy containing only a fraction of material which is relevant when learning specific concepts. For example, comparing `yt-direct` and `yt-refined` in Fig. (**4.1**) it can be seen that noise is present in the used dataset (`yt-direct`) and that it consists of title frames, non-relevant parts of the video clip or different interpretation of the semantic concept e.g. airplanes, which are not flying or soccer interviews instead of persons playing soccer. Obviously, having such frames in the training data will lead to classifiers which are less accurate as compared to training on *clean* web video.

In the next experiment, the effect of label noise present in web video datasets is investigated in the context of domain adaptation. The question to be answered is if label noise has a stronger influence on system performance than domain change.

4.4.1 Setup

The experiments were performed with systems comparable to the *Adapt-PAMIR* run i.e. the same setup and adaptation fractions α were used. But now source classifier are trained separately on two different training datasets: `yt-direct`, a noisy web video dataset, and `yt-refined`, a manually refined dataset. Both source domain classifiers are now adapted to target domains with appropriate sizes of α and detector performance is measured.

These experiments are evaluated on two different subsets of the 20 concepts defined in App. (**A**). For `tv7` the subset consists of the 12 concepts as defined by the `yt-refined` dataset. However, for the `tv5` evaluation runs an intersection of the concepts in `yt-refined` and `tv5` is used, leading to a subset of 7 concepts containing: “airplane flying”, “boat ship”, “bus”, “cityscape”, “classroom”, “demonstration” and “person playing soccer”.

Table 4.7: Results in MAP of domain adaptation runs ($n = 34,000$) on `tv7` averaged over all concepts for all values of α. First, initially trained on noisy web video and second initially trained on filtered web video.

Fraction α	**0.0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1.0**
raw data*	0.048	0.122	0.138	0.146	0.144	0.147	0.145	0.144	0.149	0.149	0.150
raw data$^+$	0.048	0.109	0.123	0.131	0.128	0.128	0.125	0.126	0.131	0.131	0.132
filtered data*	0.065	0.124	0.144	0.147	0.148	0.150	0.147	0.146	0.152	0.152	0.153
filtered data$^+$	0.065	0.112	0.127	0.132	0.133	0.132	0.130	0.130	0.134	0.135	0.135

$^*oracle$, ^+fixed

Table 4.8: Results in MAP of domain adaptation runs ($n = 37,000$) on `tv5` averaged over all concepts for all values of α. First, initially trained on noisy web video and second initially trained on filtered web video.

Fraction α	**0.0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1.0**
raw data*	0.008	0.036	0.037	0.039	0.043	0.045	0.051	0.053	0.051	0.053	0.063
raw data$^+$	0.008	0.033	0.033	0.037	0.042	0.044	0.050	0.052	0.049	0.050	0.060
filtered data*	0.013	0.036	0.038	0.038	0.043	0.043	0.050	0.051	0.052	0.052	0.059
filtered data$^+$	0.013	0.034	0.036	0.036	0.041	0.042	0.047	0.049	0.050	0.050	0.057

$^*oracle$, ^+fixed

4.4.2 Results

Results are illustrated in Fig. (**4.9**) showing an evaluation on `tv7` (left) and `tv5` (right). First, for both datasets it can be observed that classifiers trained on `yt-refined` start with an higher MAP compared to the `yt-direct` trained source classifier. For `tv7` this makes a difference of 1.7% (a relative loss of 30%), whereas the difference on `tv5` is much smaller, namely 0.5% (a relative loss of 40%) showing that in the case of no adaptation, label noise is a significant problem for web-based concept detectors. However, when performing adaptation the advantage of having a *clean* training dataset on the source domain is neglectable. A similar system performance after adaptation is observed for both runs, the one initially trained on `yt-direct` and the one initially trained on `yt-refined`. This observation can be made for `tv7` and for `tv5`. Considering performance in the context of the used adaptation fraction α as shown in Tab. (**4.7**) and Tab. (**4.8**), it can be seen that label noise becomes more irrelevant when performing adaptation with increasing values of α i.e. using larger adaptation datasets.

When analyzing concept specific performance there are exception of this generalization e.g. for full domain adaptation ($\alpha = 1.0$) on `tv7` and for the concept "airplane flying", the best results are achieved when initially trained with `yt-refined` leading to an improvement over `yt-direct` of 2.8% (a relative gain of 15%). The same could be observed for the concept "bus" with an improvement of 2.2% (a relative gain of 70%) and for the concept "female human face closeup" with an improvement of 1.1% (a relative gain of 25%). Nevertheless, the domain adaptation problem has a much stronger overall impact on web-based concept detector performance than the label noise problem.

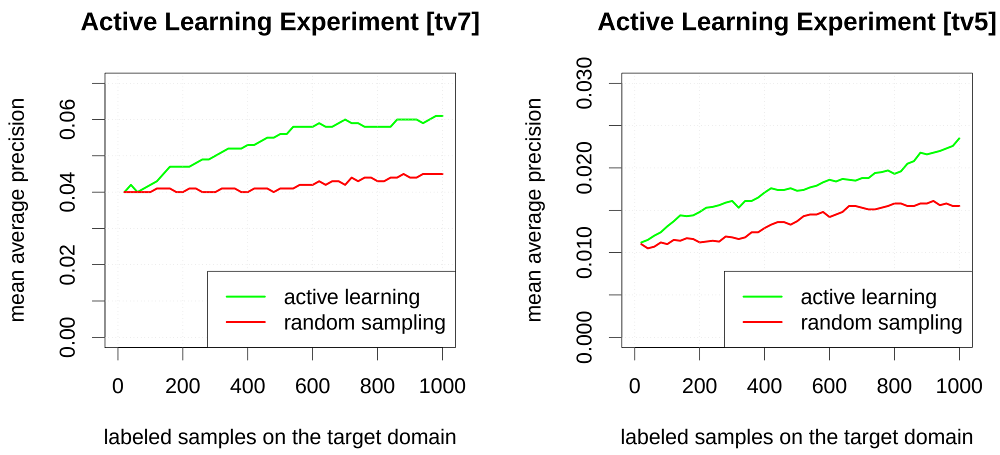


Figure 4.10: Results on the active learning extension to domain adaptation for `tv7` (left) and for `tv5` (right). As illustrated, active learning with most relevant sampling outperforms random sampling as a baseline on both target domains, demonstrating its potential for domain adaptation when only very few samples are available from the target domain

4.5 Domain Adaptation extended by Active Learning

Finally, the extension of domain adaptation with active learning is evaluated. One major benefit of the proposed domain adaptation method is its efficiency. For one concept, adaptation (about $50,000$ samples from `yt-direct` and about $34,000$ samples from `tv7.devel`) with the method takes on average 76 sec. for training and 64 sec. for adaptation (SVM: 16540 sec. for training on `tv7.devel`) whereas testing (about $34,000$ samples form `tv7.test`) takes on average less than 1 sec. (SVM: 3509 sec.).

Being able to retrain classifiers this efficient allows to incorporate user feedback within the adaptation process. Such a domain adaptation extension was proposed in Sec. (**4.5**) and is evaluated in this section for the following three active learning sample selection methods: *random sampling*, *most relevant sampling* and *uncertainty sampling*.

4.5.1 Setup

A source classifier is trained on `yt-direct` and adapted by simultaneously acquiring labels from the target domain. Again, both target domains – `tv7` and `tv5` – are evaluated in separate runs. Due to efficiency, $1,000$ samples out of the available $34,000$ from `tv7` and $37,000$ from `tv5` are annotated. For each active learning cycle 20 samples are selected for human annotation. The hyper-parameter of *Adapt-PAMIR* is set to the same values $\mathcal{C} = 0.001$ as before and the source classifier was trained as previously with $i = 10^6$ iterations. Note that annotating at most only $1,000$ samples is both realistic and very challenging as the number of positive samples in the given target domains is very sparse.

4.5.2 Results

Results of the active learning vs. random sampling are illustrated in Fig. (**4.10**). Performance is plotted against the number of annotated samples. When starting adaptation the

Table 4.9: Results (AvgP) of domain adaptation with 1000 manually annotated samples for the `tv7` and `tv5` datasets as being evaluated by active learning most relevant sampling and uncertainty sampling vs. random sampling runs.

		`tv7`			`tv5`	
Concept	**Random**	**Most Rel.**	**Uncertainty**	**Random**	**Most Rel.**	**Uncertainty**
airplane flying	0.062	**0.090**	0.036	0.015	0.020	**0.029**
boat ship	0.037	**0.120**	0.050	0.008	**0.011**	0.008
bus	0.008	**0.016**	0.012	**0.006**	**0.006**	0.005
chair	**0.007**	**0.007**	**0.007**	-	-	-
cityscape	0.078	**0.129**	0.080	**0.007**	0.005	0.006
classroom	0.018	**0.020**	0.007	**0.009**	**0.009**	0.006
demo. or protest	0.084	**0.103**	0.081	0.018	**0.024**	0.022
doorway	**0.007**	**0.007**	0.006	-	-	-
female human face closeup	0.010	**0.034**	0.022	-	-	-
hand	0.043	**0.053**	0.049	**0.003**	**0.003**	**0.003**
infant	0.036	**0.069**	0.042	**0.002**	**0.002**	**0.002**
nighttime	0.069	**0.101**	0.090	0.102	**0.138**	0.067
people dancing	0.007	**0.009**	0.007	-	-	-
person eating	0.020	**0.066**	0.005	-	-	-
person playing a musical instrument	0.026	0.013	**0.029**	-	-	-
person playing soccer	0.168	0.087	**0.229**	0.003	**0.023**	0.006
person riding a bicycle	0.096	**0.218**	0.018	-	-	-
singing	0.018	**0.028**	0.021	**0.002**	**0.002**	**0.002**
telephone	**0.002**	**0.002**	**0.002**	**0.003**	0.002	0.002
traffic intersection	0.015	**0.055**	0.014	-	-	-
MAP	0.041	**0.061**	0.040	0.015	**0.023**	0.014

performance is equal to the source domain classifier trained on `yt-direct`. With increasing amounts of samples being annotated, domain adaptation improves system performance. Thereby active learning outperforms a random selection of manually annotated samples, which can be explained by the fact that this approach better finds positive labels for a concept to be learned.

In such a situation, a robust source domain classifier provides valuable samples for domain adaptation leading to significant improvements of the detector. For example, with only $1,000$ samples and *most relevant* sampling an MAP of 6.1% can be reached for the `tv7` dataset compared to initial source classifier performance of 4.1%. Considering the `tv5` dataset, an improvement from 1.5% to 2.3% was observed when using *most relevant* sampling.

When considering per concept results (Tab. (**4.9**)) we can see that classification improvements are very concept dependent e.g. for `tv7` and “airplane flying” a similar MAP is reached by only labeling $1,000$ *most relevant* selected samples instead of $12,000$ *randomly selected* samples. However, concepts with a high redundancy (e.g. “person eating”) gain more from a source classifier independent selection (*random sampling*) than from active learning (*most relevant selection*).

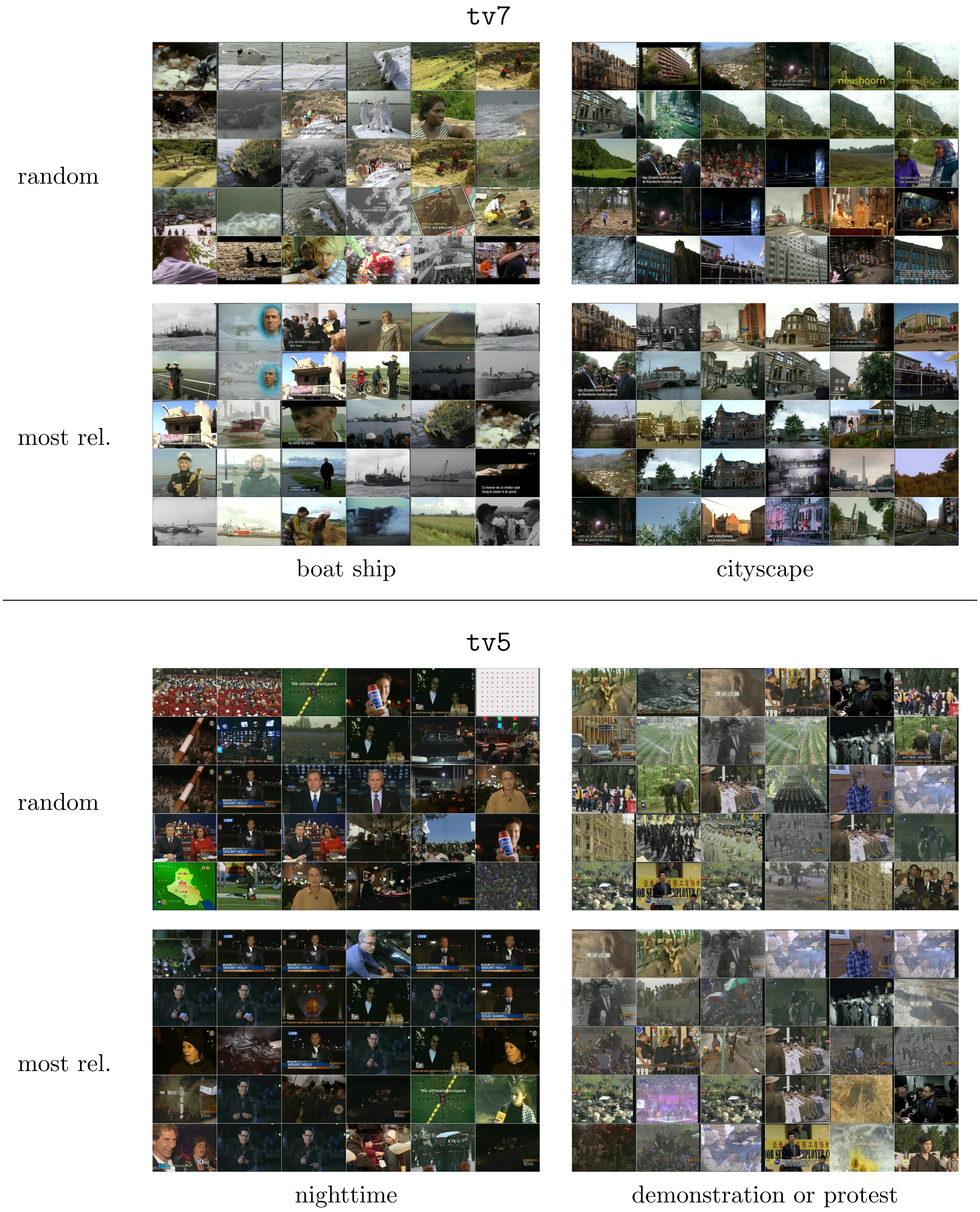


Figure 4.11: Visualization of classifiers adapted on samples selected randomly (top row) vs. adapted on samples selected by most relevant sampling (bottom row).

A visualization of system performance after adaptation with $1,000$ selected samples from the target domain is illustrated in Fig (**4.11**). The image mosaics display top ranked keyframes provides by the classifier adapted on *random sampling* (top row) and *most relevant sampling* (bottom row). As seen *most relevant sampling* improves adaptation for the concepts "boat ship", "cityscape" and "nighttime" by identifying positive samples more successfully than a random exploration of the target domain. However, having an already strong source domain classifier reduces the influence of active learning for adaptation. This can be observed for the concept "demonstration or protest", where classification mosaics for both sample selection strategies are similar.

Chapter 5

Discussion

In the final chapter of this thesis a conclusion and a future outlook is presented. This includes a summary of major experimental results and the most important insights. Additionally, the second part of this chapter suggests further direction of improvement related to the introduced approach and outlines possible future work in the investigated area of research.

5.1 Conclusion

In this thesis the domain change problem was addressed in the context of using web video as source domain for visual concept detector training. Classifiers, trained on web video as an alternative, freely available source for concept learning experience a significant performance degradation when applied on specific target domains. This impact can cause a loss of up to 60% as shown for documentary video and 80% as shown for broadcast news video. Especially in the context of web video as source domain it was shown that the domain change problem influences detection accuracy more than label noise. This observation demonstrates a great importance for cross-domain learning techniques aiming to solve this problem.

To overcome this problem, an efficient framework for domain adaptation was proposed. In contrast of previous work on cross-domain learning, adaptation is conducted in an online learning fashion. The framework trains source classifiers based on web video and adapts them to defined target domain in utilizing a small fraction of labeled data from the target domain. In quantitative experiments, it was demonstrated that domain adaptation on documentary video can be performed successfully and improves system performance from 4.1% to 13.0% (a gain of 225%) also outperforming SVMs trained on the target domain. On broadcast news video the performance of adapted classifiers could be improved from 1.0% to 5.7% (a gain of 470%). However, on this dataset the proposed domain adaptation approach was not able to outperform SVMs trained on the target domain. Facing the practical situation where no labeled adaptation data is available, an extension of the proposed method was introduced, which employs active learning strategies to support domain adaptation in providing target domain samples. Here, a simultaneous domain adaptation and label acquisition can be performed leading to an increasing adaptation dataset. This way, with the help of a proper sample selection method, domain adaptation can also be performed on very few samples from the target domain.

As shown in the experiments, successful domain adaptation depends strongly on robust source classifier. Such robustness of detection depends on the quality of web video utilized for training. Furthermore, it was shown that –for several concepts – web-based source classifiers can utilize their source domain knowledge during adaptation, which leads to adapted classifiers outperforming the ones purely trained on the target domain. However, it was also demonstrated that web-based source classifier can experience a negative transfer during adaptation i.e. detector performance was harmed by target domain adaptation.

Summarizing, the key questions from Sec. (**1.4**) have been answered throughout this thesis as following:

1. Social-tagged web video can indeed improve detector performance compared to training only on domain-specific data.
2. The associated label noise of web video does not affect domain adaptation significantly. The opposite is observed i.e. the domain change problem is identified as being the more serious cause for detector performance degradation when employing web video as source domain.
3. An efficient, light-weight domain adaptation approach was presented in this thesis which also was extended to incorporate used feedback in an interactive fashion.
4. This user feedback, which was realized as domain adaptation supported by active learning was proofed to be an efficient approach for adaptation when dealing with only few given samples from the target domain.

5.2 Future Work

In the context of this thesis, future work might focus on several different directions. Open questions which demand more investigation are *whether* an adaptation of a source classifier for a specific concept is necessary or not as seen that web video in some cases has the potential of producing more robust source classifiers than the ones trained on the target domain. Moreover, it is required to answer this question by minimizing the label information from the target domain. In the specific case of the proposed online learning algorithm, this question rephrases as: *when to stop adaptation*? Here, more robust early stopping rules could lead to performance results being as good as the oracle runs.

When it comes to practical scenarios, future work may focus on novel active learning sample selection methods. In particular, the unlabeled structure of the target domain might provide valuable clues for sample selection strategies. For example, using clustering information in combination with active learning could prevent to label redundant samples belonging to the same cluster. Other ideas, as already suggested, could be based on loss minimization or disagreement of classifier [115].

Further, domain adaptation could also be performed by employing web video from different web portals, each being one source domain. In such a setup, the most valuable source samples might be chosen to train the best fitting source classifier before adaptation. The selection of such samples could be based on the target domain knowledge and further improve domain adaptation.

Finally, a similar question is, how to download only “good” training samples from web portals like YouTube. So far, download of web video from such portals depends on the retrieval mechanism provided by the web portal. This often requires a mapping of concepts to keywords, which is usually done manually. Optimizing this mapping and therefore the retrieval of source domain data could further reduce the gap between the source domain and target domain.

Appendix A

TRECVID Concept Definitions

Clear and complete concept definitions are mandatory for a performance evaluation like TRECVID. This appendix lists the given NIST TRECVID feature definitions which serves as guidelines for all assessors during the collaborative annotation effort [5]. It is to mention that in this work the terms “concept”, “feature” and “tag” are treated equivalent.

While the semantic meaning of a concept might be blurry and depended strongly on the interpretation of the user, NIST defined three general rules how to annotate, which are given to the assessors during result evaluation. This judgment rules should also be taken into consideration when designing and building of participating systems. The judgment rules are [82]:

1. Features are meant to describe the presence or absence of *video* of some target person, place, thing, activity, etc., *not information* about that target. So, for example, video just of someone talking about x is not by itself sufficient to assert that the feature x is true with respect to the video

2. When a feature definition says a shot must contain x, that is short for “contain x to a degree sufficient for x to be recognizable as x to a human”. This means among other things that unless explicitly stated, partial visibility or audibility may suffice.

3. The fact that a segment contains video of physical objects *representing* the feature target, such as photos, paintings, models, or toy versions of the feature target, should NOT be grounds for judging the segment relevant/true. Containing video of the target within the video segment may be grounds for doing so.

In Tab. (**A**) all 20 TRECVID 2009 concept definitions are listed. In the 2009 evaluation 10 concepts are newly taken into the pool and 10 concepts form a subset of the 2008 tested ones. The old concepts are marked with a * and their definition nor their number have changed.

Table A.1: Concept definitions for the used features in the TRECVID 2009 evaluation.

TRECVID Number	**Concept Name**	**Definition**
001	classroom	*a school, university-style classroom scene. One or more students must be visible. A teacher and teaching aids (e.g. blackboard) may or may not be visible.

TRECVID Number	**Concept Name**	**Definition**
002	chair	a seat with four legs and a back for one person
003	infant	a very small child, crawling, lying down, or being held, with no evidence it can walk
004	traffic intersection	crossing of two roads or paths with some human and/or vehicular traffic visible
005	doorway	an opening you can walk through into a room or building
006	airplane flying	external view of a heavier than air, fixed-wing aircraft in flight - gliders included. NOT balloons, helicopters, missiles, and rockets
007	person playing a musical instrument	both player and instrument visible
008	bus	*external view of a large motor vehicle on tires used to carry many passengers on streets, usually along a fixed route. NOT vans and SUVs
009	person playing soccer	need not be teams or on a dedicated soccer field
010	cityscape	*a view of a large urban setting, showing skylines and building tops. NOT just street-level views of urban life
011	person riding a bicycle	a bicycle has two wheels; while riding, both feet are off the ground and the bicycle wheels are in motion
012	telephone	*any kinds of telephone, but more than just a headset must be visible.
013	person eating	putting food or drink in his/her mouth
014	demonstration or protest	*Demonstration_Or_Protest: an outdoor, public exhibition of disapproval carried out by multiple people, who may or may not be walking, holding banners or signs
015	hand	*a close-up view of one or more human hands, where the hand is the primary focus of the shot.
016	people dancing	one or more, not necessarily with each other
017	nighttime	*a shot that takes place outdoors at night. NOT sporting events under lights
018	boat ship	*exterior view of a boat or ship in the water, e.g. canoe, rowboat, kayak, hydrofoil, hovercraft, aircraft carrier, submarine, etc.
019	female human face closeup	closeup of a female human's face (face must clearly fill more than 1/2 of height or width of a frame but can be from any angle and need not be completely visible)
020	singing	*one or more people singing - singer(s) visible and audible, solo or accompanied, amateur or professional

* *Same definition as used in the TRECVID 2008 evaluation*

Appendix B

YouTube Query Definitions

This section provides the parameter used for video data download from YouTube[1]. The YouTube API offers several query parameter according to which YouTube retrieves videos. For retrieval of the used material, two parameter were used: “tag” and “category”. Tab. (**B.1**) provides the used query formulation as well as the additional category identifier.

Table B.1: Query definitions for YouTube video downloads regarding the 20 concepts used in the experiments

TRECVID Number	**Concept Name**	**YouTube Query***	**YouTube Category***
001	classroom	classroom school -secret	-
002	chair	office chair -wheel -trailer, bürostuhl	howto&style
003	infant	infant baby, kleine babys	people&blog
004	traffic intersection	traffic intersection, strassen kreuzung	autos&vehicles
005	doorway	doorway doors and gates, türen öffnen	entertainment
006	airplane flying	airplane flying -jefferson -indoor -school -kids	autos&vehicles
007	person playing a musical instrument	playing instrument, learn to play instrument	music
008	bus	bus -van -suv -vw -ride	autos&vehicles
009	person playing soccer	people playing soccer, fussball spielen	sports

* *Values used for YouTube API calls*

[1] http://code.google.com/apis/youtube/

TRECVID Number	**Concept Name**	**YouTube Query***	**YouTube Category***
010	cityscape	cityscape -slideshow -emakina	travel&places
011	person riding a bicycle	riding bicycle, fahrrad fahren	sports
012	telephone	phone device	-
013	person eating	food eating contest, essen und kochen	entertainment
014	demonstration or protest	protesting	-
015	hand	hand draft	-
016	people dancing	people dancing, learn to dance	people&blogs, sports
017	nighttime	by night	travel&places
018	boat ship	ship queen, ship freedom, ship royal	autos&vehicles
019	female human face closeup	female videoblog girl makeup	people&blog howto&style
020	singing	singing gospel, singing choire	-

* *Values used for YouTube API calls*

Appendix C

Detailed Result Tables

This appendix contains detail result listings (per concept level) for the performed experiments on the S&V target domain (`tv7`) and the LDC News target domain (`tv5`).

C.1 Details Results of Domain Adaptation: S&V

Table C.1: Results (AvgP) of domain adaptation runs on `tv7` per concept for $\alpha = \mathbf{0.1}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.062	0.033	0.069	0.037	0.069	0.063
boat ship	0.135	0.086	0.063	0.053	0.112	0.093
bus	0.006	0.015	0.016	0.013	0.000	0.015
chair	0.010	0.009	0.008	0.007	0.005	0.008
cityscape	0.173	0.131	0.102	0.089	0.102	0.141
classroom	0.014	0.009	0.006	0.007	0.001	0.014
demo. or protest	0.008	0.006	0.104	0.079	0.018	0.056
doorway	0.005	0.009	0.003	0.005	0.005	0.012
female human face closeup	0.010	0.009	0.035	0.027	0.010	0.028
hand	0.067	0.062	0.065	0.060	0.053	0.069
infant	0.004	0.012	0.022	0.042	0.000	0.035
nighttime	0.122	0.085	0.101	0.090	0.100	0.117
people dancing	0.022	0.007	0.006	0.006	0.013	0.008
person eating	0.034	0.409	0.005	0.005	0.178	0.334
person playing a musical instrument	0.011	0.011	0.040	0.030	0.010	0.026
person playing soccer	0.022	0.054	0.402	0.263	0.409	0.201
person riding a bicycle	0.310	0.285	0.043	0.016	0.067	0.285
singing	0.041	0.027	0.021	0.022	0.021	0.034
telephone	0.007	0.007	0.002	0.002	0.001	0.004
traffic intersection	0.383	0.332	0.164	0.018	0.231	0.339
MAP	0.073	0.080	0.064	0.043	0.070	0.094

Table C.2: Results (AvgP) of domain adaptation runs on `tv7` per concept for $\alpha = \mathbf{0.2}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.062	0.044	0.059	0.040	0.059	0.055
boat ship	0.112	0.131	0.068	0.058	0.133	0.124
bus	0.003	0.005	0.012	0.013	0.000	0.010
chair	0.010	0.007	0.008	0.007	0.005	0.007
cityscape	0.215	0.153	0.109	0.100	0.110	0.163
classroom	0.017	0.009	0.006	0.007	0.001	0.013
demo. or protest	0.007	0.015	0.110	0.080	0.027	0.073
doorway	0.006	0.010	0.003	0.005	0.005	0.012
female human face closeup	0.007	0.008	0.037	0.023	0.006	0.016
hand	0.107	0.082	0.082	0.068	0.062	0.088
infant	0.045	0.013	0.031	0.044	0.000	0.038
nighttime	0.109	0.119	0.101	0.092	0.101	0.132
people dancing	0.004	0.005	0.006	0.006	0.012	0.006
person eating	0.448	0.443	0.018	0.006	0.269	0.436
person playing a musical instrument	0.011	0.019	0.034	0.031	0.010	0.026
person playing soccer	0.129	0.064	0.381	0.264	0.372	0.166
person riding a bicycle	0.373	0.324	0.075	0.018	0.082	0.316
singing	0.023	0.022	0.023	0.024	0.023	0.029
telephone	0.029	0.009	0.002	0.002	0.001	0.006
traffic intersection	0.421	0.373	0.269	0.026	0.282	0.398
MAP	0.107	0.093	0.072	0.046	0.078	0.107

Table C.3: Results (AvgP) of domain adaptation runs on `tv7` per concept for $\alpha = \mathbf{0.3}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.062	0.070	0.074	0.042	0.074	0.095
boat ship	0.175	0.116	0.073	0.065	0.134	0.122
bus	0.021	0.034	0.012	0.013	0.000	0.037
chair	0.018	0.010	0.008	0.006	0.003	0.008
cityscape	0.228	0.171	0.125	0.110	0.125	0.179
classroom	0.009	0.010	0.005	0.008	0.000	0.018
demo. or protest	0.002	0.010	0.104	0.082	0.021	0.071
doorway	0.018	0.011	0.003	0.006	0.003	0.022
female human face closeup	0.010	0.012	0.038	0.027	0.011	0.023
hand	0.112	0.098	0.087	0.075	0.065	0.097
infant	0.021	0.007	0.034	0.043	0.000	0.036
nighttime	0.108	0.124	0.107	0.096	0.107	0.137
people dancing	0.015	0.007	0.006	0.006	0.018	0.007
person eating	0.441	0.446	0.036	0.007	0.304	0.443
person playing a musical instrument	0.011	0.025	0.041	0.031	0.011	0.027
person playing soccer	0.179	0.078	0.376	0.263	0.383	0.156
person riding a bicycle	0.336	0.320	0.106	0.020	0.090	0.311
singing	0.039	0.027	0.025	0.026	0.025	0.030
telephone	0.011	0.008	0.002	0.002	0.001	0.006
traffic intersection	0.414	0.367	0.290	0.033	0.321	0.398
MAP	0.112	0.098	0.078	0.048	0.085	0.112

Table C.4: Results (AvgP) of domain adaptation runs on `tv7` per concept for $\alpha = \mathbf{0.4}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.062	0.059	0.077	0.046	0.077	0.085
boat ship	0.127	0.116	0.075	0.065	0.144	0.113
bus	0.025	0.018	0.013	0.013	0.000	0.016
chair	0.009	0.010	0.009	0.006	0.006	0.002
cityscape	0.225	0.188	0.135	0.117	0.136	0.188
classroom	0.019	0.011	0.005	0.008	0.000	0.016
demo. or protest	0.030	0.012	0.116	0.077	0.030	0.066
doorway	0.024	0.012	0.003	0.007	0.004	0.014
female human face closeup	0.036	0.011	0.038	0.028	0.036	0.029
hand	0.137	0.118	0.097	0.085	0.067	0.118
infant	0.007	0.013	0.031	0.045	0.000	0.037
nighttime	0.119	0.130	0.105	0.099	0.105	0.141
people dancing	0.007	0.007	0.006	0.006	0.015	0.008
person eating	0.445	0.442	0.039	0.007	0.283	0.435
person playing a musical instrument	0.017	0.024	0.033	0.031	0.011	0.028
person playing soccer	0.139	0.066	0.388	0.263	0.371	0.147
person riding a bicycle	0.368	0.321	0.116	0.022	0.104	0.324
singing	0.035	0.025	0.024	0.028	0.024	0.031
telephone	0.015	0.008	0.002	0.002	0.002	0.007
traffic intersection	0.431	0.396	0.326	0.041	0.320	0.424
MAP	0.114	0.099	0.082	0.050	0.087	0.112

Table C.5: Results (AvgP) of domain adaptation runs on `tv7` per concept for $\alpha = \mathbf{0.5}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.097	0.094	0.079	0.050	0.079	0.129
boat ship	0.176	0.118	0.096	0.069	0.143	0.118
bus	0.013	0.023	0.015	0.014	0.000	0.027
chair	0.008	0.010	0.008	0.006	0.007	0.001
cityscape	0.221	0.203	0.145	0.127	0.146	0.207
classroom	0.016	0.014	0.005	0.008	0.000	0.018
demo. or protest	0.039	0.015	0.127	0.076	0.039	0.061
doorway	0.024	0.012	0.003	0.007	0.005	0.014
female human face closeup	0.020	0.012	0.038	0.029	0.021	0.024
hand	0.153	0.122	0.112	0.093	0.076	0.122
infant	0.023	0.019	0.026	0.047	0.001	0.058
nighttime	0.135	0.135	0.113	0.100	0.103	0.145
people dancing	0.010	0.007	0.006	0.006	0.015	0.007
person eating	0.420	0.439	0.096	0.008	0.291	0.432
person playing a musical instrument	0.015	0.027	0.037	0.031	0.011	0.028
person playing soccer	0.160	0.070	0.377	0.264	0.320	0.136
person riding a bicycle	0.313	0.305	0.132	0.025	0.117	0.298
singing	0.035	0.023	0.027	0.031	0.027	0.026
telephone	0.018	0.011	0.002	0.002	0.002	0.008
traffic intersection	0.329	0.338	0.303	0.050	0.371	0.343
MAP	0.112	0.100	0.088	0.052	0.087	0.112

Table C.6: Results (AvgP) of domain adaptation runs on `tv7` per concept for $\alpha = \mathbf{0.6}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.072	0.084	0.073	0.051	0.073	0.114
boat ship	0.127	0.117	0.095	0.072	0.148	0.117
bus	0.072	0.032	0.019	0.014	0.000	0.022
chair	0.009	0.010	0.009	0.007	0.006	0.037
cityscape	0.221	0.200	0.147	0.133	0.147	0.204
classroom	0.021	0.016	0.006	0.008	0.000	0.021
demo. or protest	0.012	0.009	0.116	0.073	0.028	0.060
doorway	0.041	0.016	0.004	0.007	0.004	0.018
female human face closeup	0.027	0.018	0.039	0.032	0.026	0.032
hand	0.151	0.132	0.117	0.097	0.079	0.128
infant	0.044	0.008	0.032	0.045	0.000	0.032
nighttime	0.116	0.126	0.113	0.102	0.113	0.137
people dancing	0.012	0.007	0.006	0.006	0.015	0.007
person eating	0.445	0.434	0.188	0.008	0.287	0.423
person playing a musical instrument	0.023	0.023	0.040	0.031	0.011	0.028
person playing soccer	0.170	0.069	0.379	0.262	0.346	0.129
person riding a bicycle	0.288	0.312	0.116	0.026	0.104	0.308
singing	0.047	0.024	0.030	0.032	0.030	0.028
telephone	0.015	0.013	0.003	0.002	0.003	0.010
traffic intersection	0.439	0.341	0.323	0.050	0.371	0.357
MAP	0.118	0.100	0.093	0.053	0.090	0.111

Table C.7: Results (AvgP) of domain adaptation runs on `tv7` per concept for $\alpha = \mathbf{0.7}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.059	0.077	0.082	0.054	0.082	0.103
boat ship	0.163	0.123	0.088	0.074	0.151	0.124
bus	0.029	0.032	0.013	0.014	0.000	0.033
chair	0.004	0.010	0.008	0.006	0.006	0.009
cityscape	0.248	0.206	0.166	0.140	0.166	0.206
classroom	0.018	0.014	0.007	0.008	0.001	0.019
demo. or protest	0.008	0.014	0.111	0.071	0.025	0.063
doorway	0.031	0.012	0.003	0.009	0.004	0.014
female human face closeup	0.017	0.015	0.036	0.031	0.017	0.033
hand	0.145	0.136	0.126	0.104	0.080	0.136
infant	0.011	0.007	0.023	0.047	0.001	0.032
nighttime	0.135	0.140	0.123	0.104	0.123	0.145
people dancing	0.009	0.007	0.006	0.006	0.011	0.006
person eating	0.451	0.424	0.149	0.008	0.341	0.405
person playing a musical instrument	0.021	0.032	0.039	0.031	0.012	0.028
person playing soccer	0.237	0.081	0.379	0.261	0.339	0.128
person riding a bicycle	0.303	0.321	0.165	0.026	0.098	0.317
singing	0.042	0.024	0.031	0.035	0.031	0.027
telephone	0.025	0.016	0.003	0.002	0.003	0.010
traffic intersection	0.422	0.351	0.337	0.055	0.367	0.381
MAP	0.119	0.102	0.095	0.054	0.093	0.111

Table C.8: Results (AvgP) of domain adaptation runs on `tv7` per concept for $\alpha = \mathbf{0.8}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.080	0.129	0.087	0.059	0.087	0.163
boat ship	0.192	0.128	0.094	0.077	0.150	0.125
bus	0.057	0.020	0.022	0.015	0.000	0.026
chair	0.010	0.011	0.008	0.006	0.006	0.009
cityscape	0.244	0.209	0.160	0.146	0.159	0.211
classroom	0.017	0.014	0.006	0.008	0.001	0.017
demo. or protest	0.035	0.014	0.114	0.070	0.045	0.064
doorway	0.025	0.014	0.004	0.008	0.004	0.016
female human face closeup	0.026	0.013	0.042	0.029	0.026	0.029
hand	0.144	0.141	0.128	0.108	0.078	0.137
infant	0.031	0.009	0.027	0.048	0.000	0.039
nighttime	0.136	0.136	0.115	0.105	0.115	0.143
people dancing	0.011	0.007	0.006	0.006	0.014	0.007
person eating	0.445	0.434	0.161	0.010	0.361	0.419
person playing a musical instrument	0.015	0.034	0.041	0.030	0.013	0.029
person playing soccer	0.166	0.072	0.378	0.264	0.323	0.126
person riding a bicycle	0.338	0.322	0.141	0.026	0.098	0.311
singing	0.040	0.028	0.032	0.035	0.032	0.031
telephone	0.009	0.014	0.003	0.002	0.003	0.009
traffic intersection	0.348	0.335	0.349	0.058	0.379	0.367
MAP	0.119	0.104	0.096	0.056	0.095	0.114

Table C.9: Results (AvgP) of domain adaptation runs on `tv7` per concept for $\alpha = \mathbf{0.9}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.070	0.103	0.086	0.059	0.086	0.146
boat ship	0.164	0.132	0.106	0.079	0.152	0.127
bus	0.048	0.016	0.024	0.015	0.000	0.025
chair	0.011	0.011	0.008	0.006	0.006	0.009
cityscape	0.237	0.211	0.175	0.150	0.175	0.212
classroom	0.017	0.015	0.007	0.009	0.001	0.020
demo. or protest	0.035	0.013	0.119	0.068	0.055	0.060
doorway	0.032	0.015	0.004	0.008	0.004	0.018
female human face closeup	0.032	0.020	0.044	0.035	0.032	0.038
hand	0.142	0.142	0.130	0.112	0.074	0.138
infant	0.049	0.016	0.025	0.048	0.001	0.051
nighttime	0.154	0.139	0.116	0.107	0.116	0.147
people dancing	0.010	0.006	0.007	0.006	0.015	0.007
person eating	0.441	0.422	0.197	0.011	0.397	0.411
person playing a musical instrument	0.015	0.036	0.044	0.030	0.014	0.029
person playing soccer	0.201	0.074	0.366	0.268	0.341	0.124
person riding a bicycle	0.288	0.313	0.158	0.028	0.117	0.303
singing	0.040	0.024	0.029	0.036	0.029	0.028
telephone	0.021	0.018	0.003	0.002	0.004	0.011
traffic intersection	0.395	0.333	0.336	0.058	0.356	0.384
MAP	0.120	0.103	0.099	0.057	0.098	0.114

C.2 Details Results of Domain Adaptation: LDC News

Table C.10: Results (AvgP) of domain adaptation runs on `tv5` per concept for $\alpha = \mathbf{0.1}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.074	0.063	0.079	0.021	0.036	0.079
boat ship	0.015	0.013	0.023	0.007	0.018	0.012
bus	0.006	0.004	0.007	0.005	0.000	0.009
cityscape	0.016	0.012	0.017	0.005	0.012	0.015
classroom	0.026	0.016	0.008	0.003	0.017	0.019
demo. or protest	0.021	0.023	0.026	0.026	0.040	0.020
hand	0.014	0.010	0.013	0.003	0.001	0.017
infant	0.001	0.002	0.001	0.001	0.000	0.002
nighttime	0.170	0.125	0.185	0.101	0.125	0.139
person playing soccer	0.097	0.061	0.103	0.008	0.047	0.072
singing	0.025	0.027	0.035	0.002	0.005	0.036
telephone	0.018	0.004	0.004	0.002	0.002	0.010
MAP	0.041	0.030	0.042	0.015	0.026	0.036

Table C.11: Results (AvgP) of domain adaptation runs on `tv5` per concept for $\alpha = \mathbf{0.2}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.101	0.067	0.123	0.026	0.044	0.078
boat ship	0.023	0.013	0.032	0.008	0.019	0.012
bus	0.009	0.007	0.013	0.005	0.000	0.007
cityscape	0.017	0.007	0.015	0.005	0.010	0.008
classroom	0.031	0.036	0.021	0.003	0.023	0.045
demo. or protest	0.021	0.022	0.018	0.026	0.037	0.022
hand	0.011	0.009	0.014	0.004	0.001	0.013
infant	0.000	0.001	0.001	0.001	0.000	0.002
nighttime	0.248	0.154	0.204	0.123	0.174	0.160
person playing soccer	0.134	0.058	0.156	0.010	0.048	0.060
singing	0.034	0.042	0.068	0.002	0.005	0.049
telephone	0.014	0.003	0.003	0.002	0.002	0.003
MAP	0.053	0.035	0.056	0.018	0.031	0.039

Table C.12: Results (AvgP) of domain adaptation runs on `tv5` per concept for $\alpha = \mathbf{0.3}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target- SVM**	**Target- PAMIR**	**Aggr- SVM**	**Aggr- PAMIR**	**Adapt- SVM**	**Adapt- PAMIR (fixed)**
airplane flying	0.116	0.063	0.123	0.028	0.047	0.086
boat ship	0.032	0.010	0.038	0.008	0.019	0.015
bus	0.005	0.005	0.006	0.005	0.000	0.005
cityscape	0.015	0.010	0.022	0.005	0.014	0.011
classroom	0.042	0.052	0.028	0.003	0.031	0.051
demo. or protest	0.021	0.023	0.023	0.027	0.034	0.022
hand	0.018	0.012	0.023	0.005	0.012	0.011
infant	0.001	0.001	0.001	0.001	0.000	0.001
nighttime	0.254	0.152	0.243	0.141	0.192	0.155
person playing soccer	0.117	0.067	0.127	0.014	0.049	0.066
singing	0.048	0.038	0.066	0.002	0.006	0.049
telephone	0.021	0.003	0.003	0.002	0.002	0.004
MAP	0.057	0.035	0.059	0.020	0.034	0.040

Table C.13: Results (AvgP) of domain adaptation runs on `tv5` per concept for $\alpha = \mathbf{0.4}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target- SVM**	**Target- PAMIR**	**Aggr- SVM**	**Aggr- PAMIR**	**Adapt- SVM**	**Adapt- PAMIR (fixed)**
airplane flying	0.120	0.073	0.125	0.028	0.068	0.081
boat ship	0.024	0.015	0.026	0.008	0.019	0.014
bus	0.008	0.008	0.008	0.005	0.000	0.007
cityscape	0.018	0.014	0.015	0.006	0.010	0.015
classroom	0.054	0.067	0.029	0.004	0.038	0.068
demo. or protest	0.022	0.026	0.023	0.029	0.038	0.025
hand	0.024	0.022	0.027	0.006	0.010	0.018
infant	0.001	0.002	0.001	0.001	0.000	0.001
nighttime	0.271	0.155	0.270	0.147	0.195	0.162
person playing soccer	0.135	0.079	0.138	0.018	0.071	0.081
singing	0.055	0.044	0.070	0.002	0.007	0.058
telephone	0.032	0.003	0.018	0.002	0.002	0.004
MAP	0.064	0.046	0.063	0.022	0.038	0.045

Table C.14: Results (AvgP) of domain adaptation runs on `tv5` per concept for $\alpha = \mathbf{0.5}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target- SVM**	**Target- PAMIR**	**Aggr- SVM**	**Aggr- PAMIR**	**Adapt- SVM**	**Adapt- PAMIR (fixed)**
airplane flying	0.136	0.078	0.152	0.033	0.095	0.088
boat ship	0.054	0.016	0.038	0.011	0.019	0.017
bus	0.011	0.006	0.008	0.006	0.000	0.006
cityscape	0.023	0.012	0.027	0.006	0.015	0.015
classroom	0.053	0.086	0.014	0.004	0.041	0.066
demo. or protest	0.021	0.026	0.023	0.029	0.037	0.025
hand	0.021	0.013	0.022	0.006	0.014	0.013
infant	0.001	0.002	0.001	0.001	0.000	0.001
nighttime	0.279	0.173	0.267	0.157	0.193	0.183
person playing soccer	0.152	0.088	0.145	0.021	0.075	0.090
singing	0.083	0.048	0.076	0.002	0.009	0.061
telephone	0.034	0.003	0.016	0.002	0.002	0.003
MAP	0.072	0.046	0.066	0.023	0.042	0.048

Table C.15: Results (AvgP) of domain adaptation runs on `tv5` per concept for $\alpha = \mathbf{0.6}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.167	0.072	0.158	0.037	0.107	0.083
boat ship	0.036	0.015	0.033	0.012	0.019	0.014
bus	0.011	0.006	0.007	0.006	0.000	0.006
cityscape	0.026	0.019	0.029	0.006	0.015	0.022
classroom	0.078	0.118	0.031	0.004	0.041	0.110
demo. or protest	0.026	0.026	0.026	0.030	0.040	0.026
hand	0.040	0.022	0.040	0.008	0.018	0.020
infant	0.001	0.002	0.001	0.001	0.001	0.001
nighttime	0.283	0.169	0.276	0.162	0.209	0.176
person playing soccer	0.152	0.088	0.144	0.026	0.072	0.085
singing	0.087	0.040	0.079	0.002	0.010	0.048
telephone	0.040	0.003	0.022	0.002	0.002	0.003
MAP	0.079	0.048	0.071	0.025	0.044	0.050

Table C.16: Results (AvgP) of domain adaptation runs on `tv5` per concept for $\alpha = \mathbf{0.7}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.160	0.067	0.141	0.037	0.098	0.080
boat ship	0.039	0.015	0.043	0.012	0.019	0.015
bus	0.008	0.005	0.007	0.006	0.000	0.006
cityscape	0.030	0.019	0.033	0.006	0.023	0.022
classroom	0.107	0.133	0.032	0.004	0.045	0.122
demo. or protest	0.025	0.027	0.026	0.030	0.035	0.026
hand	0.032	0.017	0.036	0.008	0.018	0.017
infant	0.001	0.002	0.001	0.001	0.001	0.001
nighttime	0.284	0.170	0.276	0.165	0.219	0.179
person playing soccer	0.121	0.089	0.129	0.031	0.068	0.089
singing	0.084	0.042	0.079	0.002	0.010	0.044
telephone	0.040	0.003	0.013	0.002	0.002	0.003
MAP	0.078	0.049	0.068	0.025	0.045	0.049

Table C.17: Results (AvgP) of domain adaptation runs on `tv5` per concept for $\alpha = \mathbf{0.8}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.153	0.065	0.122	0.037	0.092	0.074
boat ship	0.044	0.018	0.043	0.013	0.021	0.017
bus	0.009	0.005	0.011	0.006	0.000	0.006
cityscape	0.030	0.021	0.033	0.006	0.015	0.023
classroom	0.107	0.127	0.054	0.004	0.047	0.112
demo. or protest	0.029	0.027	0.025	0.030	0.035	0.026
hand	0.040	0.023	0.036	0.008	0.018	0.024
infant	0.001	0.002	0.001	0.001	0.001	0.001
nighttime	0.286	0.174	0.269	0.170	0.214	0.181
person playing soccer	0.135	0.090	0.147	0.037	0.076	0.084
singing	0.090	0.031	0.096	0.002	0.012	0.037
telephone	0.050	0.003	0.018	0.002	0.002	0.003
MAP	0.081	0.049	0.071	0.026	0.047	0.049

Table C.18: Results (AvgP) of domain adaptation runs on `tv5` per concept for $\alpha = \mathbf{0.9}$. Note that results for the source domain runs can be found in Tab. (**4.2**)

Concept	**Target-SVM**	**Target-PAMIR**	**Aggr-SVM**	**Aggr-PAMIR**	**Adapt-SVM**	**Adapt-PAMIR (fixed)**
airplane flying	0.150	0.071	0.145	0.037	0.095	0.079
boat ship	0.040	0.017	0.041	0.013	0.019	0.017
bus	0.008	0.006	0.008	0.006	0.000	0.006
cityscape	0.036	0.025	0.039	0.007	0.025	0.025
classroom	0.093	0.116	0.043	0.004	0.052	0.110
demo. or protest	0.029	0.028	0.028	0.030	0.039	0.027
hand	0.046	0.021	0.040	0.008	0.020	0.019
infant	0.001	0.002	0.001	0.001	0.001	0.001
nighttime	0.284	0.172	0.275	0.166	0.212	0.180
person playing soccer	0.139	0.092	0.152	0.040	0.081	0.087
singing	0.102	0.035	0.088	0.003	0.019	0.040
telephone	0.048	0.003	0.018	0.002	0.002	0.003
MAP	0.082	0.049	0.074	0.027	0.047	0.050

Bibliography

[1] R. Aly and D. Hiemstra. Concept Detectors: How good is good enough? In *Proceedings of the ACM Int. Conf. on Multimedia*, pages 233–242, 2009.

[2] A. Amir, M. Berg, S. Chang, W. Hsu, G. Iyengar, C.-Y. Lin, M. Naphade, A. Natsev, C. Neti, H. Nock, J. Smith, B. Tseng, Y. Wu, and D. Zhang. IBM Research TRECVID-2003 Video Retrieval System. In *Proc. TRECVID Workshop (unreviewed workshop paper)*, November 2003.

[3] J. Ashley, M. Flickner, J. Hafner, D. Lee, W. Niblack, and D. Petkovic. The Query by Image Content (QBIC) System. *SIGMOD Rec.*, 24(2):475, 1995.

[4] S. Ayache and G. Quenot. Evaluation of active learning strategies for video indexing. *Signal Processing: Image Communication*, 22(7-8):692–704, 2007.

[5] S. Ayache and G. Quenot. Video Corpus Annotation using Active Learning. In *Proc. Europ. Conf. on Information Retrieval*, pages 187–198, March 2008.

[6] H. Bay, T. Tuytelaars, and L. van Gool. SURF: Speeded Up Robust Features. In *ECCV*, pages 404–417, 2006.

[7] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira. Analysis of Representations for Domain Adaptation. *Advances in Neural Information Processing Systems (NIPS)*, 19:137, 2007.

[8] J. Blitzer, R. McDonald, and F. Pereira. Domain Adaptation with Structural Correspondence Learning. In *Proceedings of Conf. on Empirical Methods in Natural Language Processing*, pages 120–128, 2006.

[9] K.M. Borgwardt, A. Gretton, M.J. Rasch, H.P. Kriegel, B. Scholkopf, and A.J. Smola. Integrating Structured Biological Data by Kernel Maximum Mean Discrepancy. *Bioinformatics*, 22(14):49, 2006.

[10] D. Borth, C. Schulze, A. Ulges, and T. Breuel. Navidgator - Similarity Based Browsing for Image & Video Databases. In *Proc. KI 2008*, pages 22–29, September 2008.

[11] D. Borth, A. Ulges, and T. Breuel. Relevance Filtering meets Active Learning: Improving Web-based Concept Detectors. In *Proc. Int. Conf. on Multimedia Information Retrieval*, March 2010.

[12] D. Borth, A. Ulges, C. Schulze, and T. Breuel. Keyframe Extraction for Video Tagging and Summarization. In *Proc. Informatiktage 2008*, pages 45–48, 2008.

[13] G. Cauwenberghs and T. Poggio. Incremental and Decremental Support Vector Machine Learning. In *Advances in Neural Information Processing Systems (NIPS)*, page 409, 2001.

[14] C.-C. Chang and C.-J. Lin. *LIBSVM: A Library for Support Vector Machines*, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[15] E.Y. Chang, S. Tong, KS Goh, and C.W. Chang. Support Vector Machine Concept-Dependent Active Learning for Image Retrieval. *IEEE Trans. Multimedia*, 2, 2005.

[16] S.-F. Chang, J. He, Y.-G. Jiang, E. El Khoury, C.-W. Ngo, A. Yanagawa, and E. Zavesky. Columbia University/Video-CityU/IRIT TRECVID2008 High-Level Feature Extraction and Interactive Video Search. In *Proc. TRECVID Workshop (unreviewed workshop paper)*, November 2008.

[17] S.-F. Chang, W. Jiang, A. Yanagawa, and E. Zavesky. Columbia University TRECVID2007 High-Level Feature Extraction. In *Proc. TRECVID Workshop (unreviewed workshop paper)*, November 2007.

[18] O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-supervised Learning*. MIT Press, Cambridge, MA, 2006.

[19] M. Chen, M. Christel, A. Hauptmann, and H. Wactlar. Putting Active Learning into Multimedia Applications: Dynamic Definition and Refinement of Concept Classifiers. In *Proc. Int. Conf. on Multimedia*, pages 902–911, November 2005.

[20] S. Choudhury, J. G. Breslin, and A. Passant. Enrichment and Ranking of the YouTube Tag Space and Integration with the Linked Data Cloud. In *Int. Semantic Web Conference*, pages 747–762, 2009.

[21] D.A. Cohn, Z. Ghahramani, and M.I. Jordan. Active Learning with Statistical Models. *Journal of Artificial Intelligence Research*, 4:129–145, 1996.

[22] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer. Online Passive-Aggressive Algorithms. *The Journal of Machine Learning Research*, 7:585, 2006.

[23] K. Crammer, M. Kearns, and J. Wortman. Learning from Multiple Sources. *The Journal of Machine Learning Research*, 9:1757–1774, 2008.

[24] H. Daumé. Frustratingly Easy Domain Adaptation. In *Annual Meeting: Association for Computational Linguistics*, volume 45, page 256, 2007.

[25] O. de Rooij, C.G.M. Snoek, and M. Worring. MediaMill: Fast and Effective Video Search using the Forkbrowser. In *Proceedings of the Int. Conf. on Content-based Image and Video Retrieval*, pages 561–562, 2008.

[26] S.J. Delany, P. Cunningham, A. Tsymbal, and L. Coyle. A Case-based Technique for Tracking Concept Drift in Spam Filtering. *Knowledge-Based Systems*, 18(4-5):187–195, 2005.

[27] T. Deselaers, D. Keysers, and H. Ney. Features for Image Retrieval: An Experimental Comparison. *Information Retrieval*, 11:77–107, 03/2008 2008.

[28] L. Duan, I.W. Tsang, D. Xu, and T.S. Chua. Domain Adaptation from Multiple Sources via Auxiliary Classifiers. In *Proceedings of the Int. Conf. on Machine Learning*, 2009.

[29] L. Duan, I.W. Tsang, D. Xu, and S.J. Maybank. Domain Transfer SVM for Video Concept Detection. In *Proceedings of the Int. Conf. on Pattern Recognition*, 2009.

[30] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results, October 2008.

[31] Y. Freund, H.S. Seung, E. Shamir, and N. Tishby. Selective Sampling using the Query by Committee Algorithm. *Machine Learning*, 28(2):133–168, 1997.

[32] U. Gargi and J. Yagnik. Solving the Label Resolution Problem in Supervised Video Content Classification. In *Proc. Int. Conf. on Multimedia Retrieval*, pages 276–282, October 2008.

[33] S.A. Golder and B.A. Huberman. Usage Patterns of Collaborative Tagging Systems. *Journal of Information Science*, 32(2):198, 2006.

[34] Gonzlez-Daz, I. et al. UC3M High Level Feature Extraction at TRECVID 2008. In *Proc. TRECVID Workshop (unreviewed workshop paper)*, 2008.

[35] D. Grangier and S. Bengio. A Discriminative Kernel-based Model to Rank Images from Text Queries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1371–1384, 2008.

[36] Z. Gu, T. Mei, X.S. Hua, J. Tang, and X. Wu. Multi-layer Multi-Instance Learning for Video Concept Detection. *IEEE Transactions on Multimedia*, 10(8):1605–1616, 2008.

[37] R. Hammoud and R. Mohr. A Probabilistic Framework of Selecting Effective Key Frames for Video Browsing and Indexing. In *Proc. Int. Workshop on Real-Time Image Sequence Analysis*, pages 79–88, August 2000.

[38] A. Hauptmann, R. Yan, and W. Lin. How many High-Level Concepts will Fill the Semantic Gap in News Video Retrieval? In *Proc. Int. Conf. Image and Video Retrieval*, pages 627–634, July 2007.

[39] A.G. Hauptmann, W.H. Lin, R. Yan, J. Yang, and M.Y. Chen. Extreme Video Retrieval: Joint Maximization of Human and Computer Performance. In *Proc. Int. Conf. on Multimedia*, pages 385–394. ACM Press New York, NY, USA, 2006.

[40] J. Huang, A.J. Smola, A. Gretton, K.M. Borgwardt, and B. Scholkopf. Correcting Sample Selection Bias by Unlabeled Data. *Advances in Neural Information Processing Systems (NIPS)*, 19:601, 2007.

[41] Cisco Systems Inc. Cisco Visual Networking Index: Forecast and Methodology, 2008-2013. available from http://www.cisco.com (retrieved: June'09), June 2009.

[42] Michael Jackson memorial draws crowds online. available from http://edition.cnn.com/2009/TECH/07/07/michael.jackson.web.traffic/ (retrieved: Jan'10), July 2009.

[43] J. Jiang. A Literature Survey on Domain Adaptation of Statistical Classifiers. 2007.

[44] W. Jiang, E. Zavesky, S.-F. Chang, and A. Loui. Cross-domain Learning Methods for High-level Visual Concept Classification. In *ICIP*, pages 161–164, 2008.

[45] Y.-G. Jiang, C.-W. Ngo, and J. Yang. Towards Optimal Bag-of-Features for Object Categorization and Semantic Video Retrieval. In *Proc. Int. Conf. Image and Video Retrieval*, pages 494–501, July 2007.

[46] Y.G. Jiang, C.W. Ngo, and S.F. Chang. Semantic Context Transfer across Heterogeneous Sources for Domain Adaptive Video Search. In *Proceedings of the ACM Int. Conf. on Multimedia*, pages 155–164. ACM, 2009.

[47] Y.G. Jiang, J. Wang, S.F. Chang, and C.W. Ngo. Domain Adaptive Semantic Diffusion for Large Scale Context-based Video Annotation. Proceedings of Int. Conf. on Computer Vision, 2009.

[48] R. Junee. Zoinks! 20 Hours of Video Uploaded Every Minute! The YouTube Blog; available from http://www.youtube.com/blog?entry=on4EmafA5MA (retrieved: May'09), May 2009.

[49] J. Kivinen, A.J. Smola, and R.C. Williamson. Online Learning with Kernels. *IEEE Transactions on Signal Processing*, 52(8):2165–2176, 2004.

[50] R. Klinkenberg and T. Joachims. Detecting Concept Drift with Support Vector Machines. In *Proceedings of the Int. Conf. on Machine Learning*, pages 487–494, 2000.

[51] J.Z. Kolter and M.A. Maloof. Dynamic Weighted Majority: An Ensemble Method for Drifting Concepts. *The Journal of Machine Learning Research*, 8:2755–2790, 2007.

[52] Georgia Koutrika, Frans Adjie Effendi, Zolt´n Gyöngyi, Paul Heymann, and Hector Garcia-Molina. Combating spam in tagging systems: An evaluation. *ACM Trans. Web*, 2(4):1–34, 2008.

[53] D. Lewis and W. Gale. A Sequential Algorithm for Training Text Classifiers. In *Proc. Int. Conf. Research and Development in Information Retrieval*, pages 3–12, July 1994.

[54] R. Lienhart. Reliable Transition Detection in Videos: A Survey and Practitioner's Guide. *Int. J. of Img. and Graph.*, 1(3):469–286, 2001.

[55] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *IJCV*, 60(2):91–110, 2004.

[56] LSCOM - Large-scale Concept Ontology for Multimedia. available from: http://www.lscom.org (retrieved: Aug'08).

[57] P. Luo, F. Zhuang, H. Xiong, Y. Xiong, and Q. He. Transfer Learning from Multiple Source Domains via Consensus Regularization. In *Proceeding of ACM Int. Conf. on Information and Knowledge Management*, pages 103–112, 2008.

[58] Y. Mansour, M. Mohri, and A. Rostamizadeh. Domain Adaptation with Multiple Sources. volume 21, pages 1041–1048, 2009.

[59] M. Naphade, J. Smith, J. Tesic, S. Chang, W. Hsu, L. Kennedy, A. Hauptmann, and J. Curtis. Large-Scale Concept Ontology for Multimedia. *IEEE MultiMedia*, 13(3):86–91, 2006.

[60] C.-W. Ngo, Y.-G. Jiang, X.-Y. Wei, W. Liu, S. Zhu, and S.-F. Chang. VIREO/DVMM at TRECVID 2009: High-Level Feature Extraction, Automatic Video Search, and Conctent-based Copy Detection. In *Proc. TRECVID Workshop*, November 2009.

[61] H.T. Nguyen and A. Smeulders. Active learning using Pre-Clustering. In *Proc. International Conf. on Machine Learning*, pages 623–630, July 2004.

[62] Online Inauguration Videos set Records. available from http://edition.cnn.com/2009/TECH/01/21/inauguration.online.video/ (retrieved: Jan'10), January 2009.

[63] P. Over, G. Awad, W. Kraaij, and A. Smeaton. TRECVID 2007 - An Overview. In *Proc. TRECVID Workshop (unreviewed workshop paper)*, November 2007.

[64] P. Over, G. Awad, T. Rose, J. Fiscus, W. Kraaij, and A. Smeaton. TRECVID 2009–Goals, Tasks, Data, Evaluation Mechanisms and Metrics. In *Proc. TRECVID Workshop*, November 2009.

[65] S.J. Pan, J.T. Kwok, and Q. Yang. Transfer Learning via Dimensionality Reduction. In *Proceedings of the AAAI Conf. on Artificial Intelligence*, pages 677–682, 2008.

[66] R. Paredes, A. Ulges, and T. Breuel. Fast Discriminative Linear Models for Scalable Video Tagging. In *Proc. Int. Conf. on Machine Learning and Applications*, 2009.

[67] C. Petersohn. Fraunhofer HHI at TRECVID 2004: Shot boundary detection system. In *TREC Video Retrieval Evaluation Online Proceedings, TRECVID*, 2004.

[68] J.C. Platt. Probabilities for SV Machines. *Advances in Neural Information Processing Systems (NIPS)*, pages 61–74, 1999.

[69] G. Qi, Y. Song, X.S. Hua, H.J. Zhang, and L.R. Dai. Video Annotation by Active Learning and Cluster Tuning. In *Computer Vision and Pattern Recognition Workshop*, pages 114–114, 2006.

[70] G.J. Qi, X.S. Hua, Y. Rui, J. Tang, T. Mei, and H.J. Zhang. Correlative Multi-Label Video Annotation. In *Proceedings of the ACM Int. Conf. on Multimedia*, page 26, 2007.

[71] R. Raina, A. Battle, H. Lee, B. Packer, and A.Y. Ng. Self-taught Learning: Transfer Learning from Unlabeled Data. In *Proceedings of Int. Conf. on Machine learning*, page 766, 2007.

[72] M.T. Rosenstein, Z. Marx, L.P. Kaelbling, and T.G. Dietterich. To Transfer or not to Transfer. In *Proceedings of NIPS 2005 Workshop on Inductive Transfer: 10 Years Later*. Citeseer, 2005.

[73] N. Roy and A. McCallum. Toward Optimal Active Learning through Sampling Estimation of Error Reduction. In *Proc. 18th International Conf. on Machine Learning*, pages 441–448, 2001.

[74] G. Salton and C. Buckley. Improving Retrieval Performance by Relevance Feedback. *Journal of the American Society for Information Science*, 41(4):288–297, 1990.

[75] G. Schohn and D. Cohn. Less is More: Active Learning with Support Vector Machines. In *In Proceedings of the Int. Conf. on Machine Learning*, 2000.

[76] B. Schölkopf and A. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.

[77] B. Settles. Active Learning Literature Survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[78] A. Setz and C. Snoek. Can Social Tagged Images Aid Concept-Based Video Search? In *Proc. Int. Conf. on Multimedia and Expo*, pages 1460–1463, 2009.

[79] HS Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proc. of the fifth annual workshop on Computational learning theory*, pages 287–294. ACM New York, NY, USA, 1992.

[80] J. Sivic and A. Zisserman. Video Google: A Text Retrieval Approach to Object Matching in Videos. In *Proc. Int. Conf. Computer Vision*, pages 1470–1477, October 2003.

[81] A. Smeaton. Large Scale Evaluations of Multimedia Information Retrieval: The TRECVid Experience. In *Proc. Int. Conf. Image and Video Retrieval*, pages 11–17, July 2005.

[82] Alan F. Smeaton, Paul Over, and Wessel Kraaij. High-Level Feature Detection from Video in TRECVid: a 5-Year Retrospective of Achievements. In *Multimedia Content Analysis, Theory and Applications*, pages 151–174. 2009.

[83] Smeaton, A. F. and Wilkins, P. and Worring, M. and de Rooij, O.". Content-based video retrieval: Three example systems from trecvid. *International Journal of Imaging Science and Technology*, 18(2-3):195–201, 2008.

[84] A. Smeulders, M. Worring, S. Santini, and A. Gupta R. Jain. Content-Based Image Retrieval at the End of the Early Years. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.

[85] C. Snoek and M. Worring. Concept-based Video Retrieval. *Foundations and Trends in Information Retrieval*, 4(2):215–322, 2009.

[86] C. Snoek, M. Worring, O. de Rooij, K. van de Sande, R. Yan, and A. Hauptmann. VideOlympics: Real-Time Evaluation of Multimedia Retrieval Systems. *IEEE MultiMedia*, 15(1):86–91, 2008.

[87] C. Snoek, M. Worring, J. van Gemert, J. Geusebroek, and A. Smeulders. The Challenge Problem for Automated Detection of 101 Semantic Concepts in Multimedia. In *Proc. Int. Conf. on Multimedia*, pages 225–226, October 2006.

[88] C.G.M. Snoek, M. Worring, and A.W.M. Smeulders. Early versus Late Fusion in Semantic Video Analysis. In *Proceedings of ACM Int. Conf. on Multimedia*, page 402. ACM, 2005.

[89] Snoek, C. et al. The MediaMill TRECVID 2009 Semantic Video Search Engine. In *Proc. TRECVID Workshop (unreviewed workshop paper)*, 2009.

[90] Y. Song, X.S. Hua, L.R. Dai, and M. Wang. Semi-automatic Video Annotation based on Active Learning with Multiple Complementary Predictors. In *Proceedings of the Int. Workshop on Multimedia Information Retrieval*, page 104, 2005.

[91] M. Sugiyama, M. Krauledat, and K.R. M
"uller. Covariate Shift Adaptation by Importance Weighted Cross Validation. *The Journal of Machine Learning Research*, 8:985–1005, 2007.

[92] JAK Suykens and J. Vandewalle. Least Squares Support Vector Machine Classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.

[93] S. Tong and E. Chang. Support Vector Machine Active Learning for Image Retrieval. In *Proc. Int. Conf. on Multimedia*, pages 107–118, September 2001.

[94] S. Tong and D. Koller. Support Vector Machine Active Learning with Applications to Text Classification. *The Journal of Machine Learning Research*, 2:45–66, 2002.

[95] Time Warner Inc. Announces Widespread Distribution of Cable TV Content Online. in Press Releases from TimeWarner Inc. ;available form http://www.timewarner.com/corp/newsroom/pr/0,20812,1906715,00.html. (retrieved: Jun'09), 2007.

[96] Video Metrix: A Comprehensive View of the Online Video Landscape. in Video Metrix from ComScore ;available form http://www.comscore.com/Products_Services/Product_Index/Video_Metrix/. (retrieved: Feb'10), 2007.

[97] A. Ulges, D. Borth, and T. Breuel. Visual Concept Learning from Weakly Labeled Web Videos (submitted for publication). In *Video Search and Mining*. Springer-Verlag, 2009.

[98] A. Ulges, M. Koch, D. Borth, and T. Breuel. Tubetagger - youtube-based concept detection. In *Proc. Int. Workshop on Internet Multimedia Mining*, Dec 2009.

[99] A. Ulges, M. Koch, C. Schulze, and T. Breuel. Learning TRECVID'08 High-level Features from YouTube^{TM}. In *Proc. TRECVID Workshop (unreviewed workshop paper)*, November 2008.

[100] A. Ulges, C. Schulze, D. Keysers, and T. Breuel. Identifying Relevant Frames in Weakly Labeled Videos for Training Concept Detectors. In *Proc. Int. Conf. Image and Video Retrieval*, pages 9–16, July 2008.

[101] A. Ulges, C. Schulze, M. Koch, and T. Breuel. The Challenge of Tagging Online Video. *Comp. Vis. Img. Underst. (submitted for publication)*, 2009.

[102] K. van de Sande, T. Gevers, and C. Snoek. A Comparison of Color Features for Visual Concept Classification. In *Proc. Int. Conf. Image and Video Retrieval*, pages 141–150, July 2008.

[103] K.E.A. Van De Sande, T. Gevers, and C.G.M. Snoek. Evaluation of Color Descriptors for Object and Scene Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[104] V.N. Vapnik. *The Nature of Statistical Learning Theory.* Springer Verlag, 2000.

[105] D. Wang, X. Liu, L. Luo, J. Li, and B. Zhang. Video Diver: Generic Video Indexing with Diverse Features. In *Proc. Int. Workshop Multimedia Information Retrieval*, pages 61–70, September 2007.

[106] H. Wang, W. Fan, P.S. Yu, and J. Han. Mining Concept-Drifting Data Streams using Ensemble Classifiers. In *Proceedings of Int. Conf. on Knowledge Discovery and Data Mining*, pages 226–235, 2003.

[107] M. Wang, X.-S. Hua, Y. Song, X. Yuan, S. Li, and H.-J. Zhang. Automatic Video Annotation by Semi-supervised Learning with Kernel Density Estimation. In *Proc. Int. Conf. on Multimedia*, pages 967–976, October 2006.

[108] P. Wu and T.G. Dietterich. Improving SVM Accuracy by Training on Auxiliary Data Sources. In *Proceedings of Int. Conf on Machine learning*, 2004.

[109] X. Wu and R. Srihari. Incorporating Prior Knowledge with Weighted Margin Support Vector Machines. In *Proceedings of Int. Conf. on Knowledge Discovery and Data Mining*, pages 326–333, 2004.

[110] R. Yan and A.G. Hauptmann. A review of text and image retrieval approaches for broadcast news video. *Information Retrieval*, 10(4):445–484, 2007.

[111] A. Yanagawa, S.-F. Chang, L. Kennedy, and W. Hsu. Columbia University's Baseline Detectors for 374 LSCOM Semantic Visual Concepts. Technical report, Columbia University, 2007.

[112] J. Yang and A. Hauptmann. (Un)Reliability of Video Concept Detection. In *Proc. Int. Conf. Image and Video Retrieval*, pages 85–94, July 2008.

[113] J. Yang and A.G. Hauptmann. A Framework for Classifier Adaptation and its Applications in Concept Detection. In *Proc. Int. Conf. on Multimedia Information Retrieval)*, 2008.

[114] J. Yang, R. Yan, and A. Hauptmann. Cross-Domain Video Concept Detection using Adaptive SVMs. In *Proc. Int. Conf. on Multimedia*, pages 188–197, September 2007.

[115] J. Yang, R. Yan, and A. Hauptmann. Learning to Adapt Across Multimedia Domains. In *ACM Int. Conf. on Multimedia*, 2007.

[116] Y,000,000,000uTube. Official YouTube Blog; available from http://youtube-global.blogspot.com/2009/10/y000000000utube.html (retrieved: Oct'09), October 2009.