The MediaMill TRECVID 2010 Semantic Video Search Engine

Draft notebook paper

C.G.M. Snoek, K.E.A. van de Sande, O. de Rooij, B. Huurnink, E. Gavves, D. Odijk,
M. de Rijke, Th. Gevers, M. Worring, D.C. Koelma, A.W.M. Smeulders
ISLA, University of Amsterdam
Amsterdam, The Netherlands
http://www.mediamill.nl

Abstract

In this paper we describe our TRECVID 2010 video retrieval experiments. The MediaMill team participated in three tasks: semantic indexing, known-item search, and instance search. The starting point for the MediaMill concept detection approach is our top-performing bag-of-words system of last year, which uses multiple color SIFT descriptors, sparse codebooks with spatial pyramids, kernel-based machine learning, and multi-frame video processing. We improve upon this baseline system by further improving its execution times for both training and classification using GPU-optimized algorithms, approximated histogram intersection kernels, and several multi-frame combination methods. Being more efficient allowed us to supplement the Internet video training collection with positively labeled examples from international news broadcasts and Dutch documentary video from the TRECVID 2005-2009 benchmarks. Our experimental setup covered a huge training set of 170 thousand keyframes and a test set of 600 thousand keyframes in total. Ultimately leading to 130 robust concept detectors for video retrieval. For retrieval, a robust but limited set of concept detectors justifies the need to rely on as many auxiliary information channels as possible. For automatic known item search we therefore explore how we can learn to rank various information channels simultaneously to maximize video search results for a given topic. To further improve the video retrieval results, our interactive known item search experiments investigate how to combine metadata search and visualization into a single interface. The 2010 edition of the TRECVID benchmark has again been a fruitful participation for the MediaMill team, resulting in the top ranking for concept detection in the semantic indexing task. Again a lot has been learned during this year's TRECVID campaign; we highlight the most important lessons at the end of this paper.

1 Introduction

Robust video retrieval is highly relevant in a world that is adapting swiftly to visual communication. Online services like YouTube and Vimeo show that video is no longer the domain of broadcast television only. Video has become the medium of choice for many people communicating via the Internet. Most commercial video search engines provide access to video based on text, as this is still the easiest way for a user to describe an information need. The indices of these search engines are based on the filename, surrounding text, social tagging, closed captions, or a speech transcript. This results in disappointing retrieval performance when the visual content is not mentioned, or properly reflected in the associated text. In addition, when the videos originate from non-English speaking countries, such as China, or the Netherlands, querying the content becomes much harder as robust automatic speech recognition results and their accurate machine translations are difficult to achieve.

To cater for robust video retrieval, the promising solutions from literature are mostly concept-based [21], where detectors are related to objects, like an *airplane flying*, scenes, like a *cityscape*, and people, like *female human face closeup*. Any one of those brings an understanding of the current content. The elements in such a lexicon of concept detectors offer users a semantic entry to video by allowing them to query on presence or absence of visual content elements. Last year we presented the *MediaMill 2009* semantic video search engine [19], which made our robust concept detection system more efficient [25,27]. We have recently shown that progress in visual concept search has doubled in just 3 years [18]. Surprisingly, the progress even holds for cross-domain visual search engines, albeit with a loss in performance compared to within-domain search engines. Rather than further pushing the envelope in terms of within-domain performance, our TRECVID 2010 experiments focus on what is needed for successful cross-domain experiments on Internet video. In particular, we improve execution times for both training and classification using GPU-optimized algorithms [27], approximated histogram intersection kernels [11], and several multi-frame combination methods. Being more efficient allowed us to supplement the Internet video training collection with positively labeled examples from international news broadcasts and Dutch documentary video from the TRECVID 2005-2009 benchmarks.

A robust but limited set of concept detectors justifies the need to rely on as many multimedia information channels

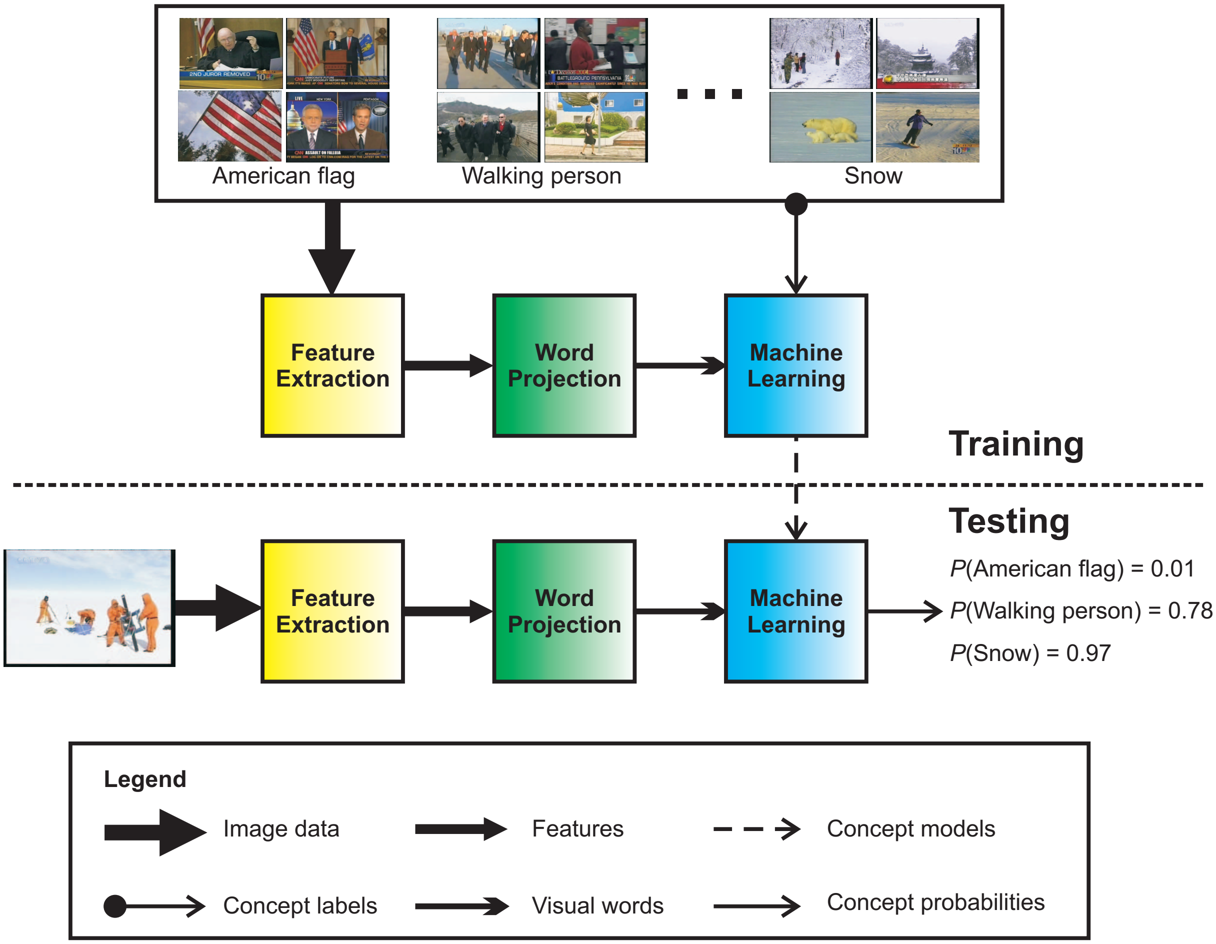

Figure 1: MediaMill TRECVID 2010 concept detection scheme, which serves as the blueprint for the organization of Section 2.

as possible for retrieval. To that end, we explore how we can learn to rank various information channels simultaneously to maximize video search results for a given topic. To improve the retrieval results further, we rely on an interacting user who combines metadata search and extensive metadata visualization into an extended version of our MediaTable [3]. Finally, we explore the situation in which a user can only query based on a limited set of example images. Taken together, the *MediaMill 2010* semantic video search engine provides users with robust semantic access to Internet video collections.

The remainder of the paper is organized as follows. We first define our semantic concept detection scheme in Section 2. Then we highlight our automatic video retrieval framework for known item search in Section 3, and for interactive video retrieval in Section 4. We summarize our efforts in the instance search pilot in Section 5.

2 Detecting Concepts in Video

We perceive concept detection in video as a combined multimedia analysis and machine learning problem. Given an n-dimensional multimedia feature vector x_i, part of a shot i [13], the aim is to obtain a measure, which indicates whether semantic concept ω_j is present in shot i. We may choose from various audiovisual feature extraction methods to obtain x_i, and from a variety of supervised machine learning approaches to learn the relation between ω_j and x_i. The supervised machine learning process is composed of two phases: training and testing. In the first phase, the optimal configuration of features is learned from the training data. In the second phase, the classifier assigns a probability $p(\omega_j|x_i)$ to each input feature vector for each semantic concept.

Our TRECVID 2010 concept detection approach builds on previous editions of the MediaMill semantic video search engine [19, 20, 23], which draws inspiration from the bag-of-words approach propagated by Schmid and her associates [7,12,34], as well as recent advances in keypoint-based color features [26] and codebook representations [28, 30]. Last year, we made the system more efficient with algorithmic refinements of the bag-of-words approach [25], a GPU implementation [27], and compute clusters. Rather than further pushing the envelope in terms of within-domain performance, our TRECVID 2010 experiments focus on what is needed for successful cross-domain experiments on Internet video. In particular, we improve execution times for both training and classification using approximated histogram intersection kernels [11], and we explore several multi-frame combination methods. We detail our generic concept detection scheme by presenting a component-wise decomposition. The components exploit a common architecture, with a standardized input-output model, to allow for semantic integration. We follow the video data as it flows through the computational process, as summarized in the general scheme of our TRECVID 2010 concept detection approach in Figure 1, and detailed per component next.

2.1 Feature Extraction

2.1.1 Spatio-Temporal Sampling

The visual appearance of a semantic concept in video has a strong dependency on the spatio-temporal viewpoint under which it is recorded. Salient point methods [24] introduce robustness against viewpoint changes by selecting points, which can be recovered under different perspectives. Another solution is to simply use many points, which is achieved by dense sampling. Appearance variations caused by temporal effects are addressed by analyzing video beyond the key frame level. By taking more frames into account during analysis, it becomes possible to recognize concepts that are visible during the shot, but not necessarily in a single key frame.

Temporal multi-frame selection In [19, 20, 22] we demonstrated that a concept detection method that considers more video content obtains higher performance over key frame-based methods. We attribute this to the fact that the content of a shot changes due to object motion, camera motion, and imperfect shot segmentation results. Therefore, we employ a multi-frame sampling strategy. To be precise, we sample up to 6 additional i-frames distributed around the (middle) key frame of each shot.

Harris-Laplace point detector In order to determine salient points, Harris-Laplace relies on a Harris corner detector. By applying it on multiple scales, it is possible to select the characteristic scale of a local corner using the Laplacian operator [24]. Hence, for each corner, the Harris-Laplace detector selects a scale-invariant point if the local image structure under a Laplacian operator has a stable maximum.

Dense point detector For concepts with many homogenous areas, like scenes, corners are often rare. Hence, for these concepts relying on a Harris-Laplace detector can be suboptimal. To counter the shortcoming of Harris-Laplace, random and dense sampling strategies have been proposed [4, 6]. We employ dense sampling, which samples an image grid in a uniform fashion using a fixed pixel interval between regions. In our experiments we use an interval distance of 6 pixels and sample at multiple scales.

Spatial pyramid weighting Both Harris-Laplace and dense sampling give an equal weight to all keypoints, irrespective of their spatial location in the image frame. In order to overcome this limitation, Lazebnik *et al.* [7] suggest to repeatedly sample fixed subregions of an image, *e.g.*,1x1, 2x2, 4x4, *etc.*, and to aggregate the different resolutions into a so called spatial pyramid, which allows for region-specific weighting. Since every region is an image in itself, the spatial pyramid can be used in combination with both the Harris-Laplace point detector and dense point sampling. Similar to [12, 19, 20] we use a spatial pyramid of 1x1, 2x2, and 1x3 regions in our experiments.

2.1.2 Visual Feature Extraction

In the previous section, we addressed the dependency of the visual appearance of semantic concepts in a video on the spatio-temporal viewpoint under which they are recorded. However, the lighting conditions during filming also play an important role. Burghouts and Geusebroek [1] analyzed the properties of color features under classes of illumination and viewing changes, such as viewpoint changes, light intensity changes, light direction changes, and light color changes. Van de Sande *et al.* [26] analyzed the properties of color features under classes of illumination changes within the diagonal model of illumination change, and specifically for data sets as considered within TRECVID.

SIFT The SIFT feature proposed by Lowe [10] describes the local shape of a region using edge orientation histograms. The gradient of an image is shift-invariant: taking the derivative cancels out offsets [26]. Under light intensity changes, *i.e.*,a scaling of the intensity channel, the gradient direction and the relative gradient magnitude remain the same. Because the SIFT feature is normalized, the gradient magnitude changes have no effect on the final feature. To compute SIFT features, we use the version described by Lowe [10].

OpponentSIFT OpponentSIFT describes all the channels in the opponent color space using SIFT features. The information in the O_3 channel is equal to the intensity information, while the other channels describe the color information in the image. The feature normalization, as effective in SIFT, cancels out any local changes in light intensity.

RGB-SIFT For the RGB-SIFT, the SIFT feature is computed for each RGB channel independently. Due to the normalizations performed within SIFT, it is equal to transformed color SIFT [26]. The feature is scale-invariant, shift-invariant, and invariant to light color changes and shift.

We compute the SIFT [10] and ColorSIFT [26] features around salient points obtained from the Harris-Laplace detector and dense sampling. For all visual features we employ a spatial pyramid of 1x1, 2x2, and 1x3 regions.

2.2 Word Projection

To avoid using all visual features in an image, while incorporating translation invariance and a robustness to noise, we follow the well known codebook approach, see *e.g.*, [6, 8, 16, 28, 30]. First, we assign visual features to discrete codewords predefined in a codebook. Then, we use the frequency distribution of the codewords as a compact feature vector representing an image frame. By using a vectorized GPU implementation [27], our codebook transform process is an order of magnitude faster for the most expensive feature compared to the standard implementation. Two important variables in the codebook representation are *codebook construction* and *codeword assignment*. Based on previous experiments, balancing accuracy and performance, we employ codebook construction using k-means clustering in combination with hard codeword assignment and a maximum of 4,096 codewords.

Kernel library Each of the possible sampling methods from Section 2.1 coupled with each visual feature extraction method from Section 2.1.2, a clustering method, and an assignment approach results in a separate visual codebook. An example is a codebook based on dense sampling of RGB-SIFT features in combination with k-means clustering and hard assignment. We collect all possible codebook combinations in a (visual) kernel library. By using a GPU implementation [27], this kernel library can be computed efficiently. Naturally, the codebooks can be combined using various configurations. Depending on the kernel-based learning scheme used, we simply employ equal weights in our experiments or learn the optimal weight using cross-validation.

2.3 Machine Learning

Learning robust concept detectors from visual features is typically achieved by kernel-based learning methods. Similar to previous years, we rely predominantly on the support vector machine framework [31] for supervised learning of semantic concepts. Here we use the LIBSVM implementation [2] with probabilistic output [9, 14]. In order to handle imbalance in the number of positive versus negative training examples, we fix the weights of the positive and negative class by estimation from the class priors on training data. While the radial basis kernel function usually performs better than other kernels, it was recently shown by Zhang *et al.* [34] that in a codebook-approach to concept detection the earth movers distance [15] and χ^2 kernel are to be preferred. For multi-frame processing of video, classifying frames at testing time becomes computationally complex using a χ^2 kernel. For this year's TRECVID we investigate the use of Histogram Intersection kernels and its efficient approximation as suggested by Maji *et al.* [11]. In general, we obtain good parameter settings for a support vector machine, by using an iterative search on both C and kernel function $K(\cdot)$ on cross validation data [29].

Episode-constrained cross-validation From all parameters q we select the combination that yields the best average precision performance, yielding q^*. We measure performance of all parameter combinations and select the combination that yields the best performance. We use a 3-fold cross validation to prevent over-fitting of parameters. Rather than using regular cross-validation for support vector machine parameter optimization, we employ an *episode-constrained* cross-validation method, as this method is known to yield a less biased estimate of classifier performance [29].

The result of the parameter search over q is the improved model $p(\omega_j|x_i, q^*)$, contracted to $p^*(\omega_j|x_i)$, which we use to fuse and to rank concept detection results.

2.4 Submitted Concept Detection Results

Rather than further pushing the envelope in terms of within-domain performance, we explore for this year's TRECVID edition what is needed for successful cross-domain experiments on Internet video. Being more efficient allowed us to supplement the Internet video training collection with positively labeled examples from international news broadcasts and Dutch documentary video from the TRECVID 2005-2009 benchmarks. For the cross-domain experiments, our experimental setup covered a huge training set of 170 thousand keyframes and a test set of 600 thousand keyframes in total. In addition to the cross-domain experiments, we study the effect of approximate intersection kernels, and multi-frame combiner functions. An overview of our submitted concept detection runs is depicted in Figure 2, and detailed next.

Run: Captain Slow The *Captain Slow* run is our multi frame baseline, using the Internet video training data only. It is based on multiple (visual) kernel libraries using SIFT, OpponentSIFT, and RGB-SIFT only, which have been applied spatio-temporally with up to 6 additional i-frames per shot in combination with an AVG rule combination. This run achieved the overall highest mean infAP in the TRECVID2010 benchmark (0.0900), with the overall highest infAP for 4 concepts: *cityscape*, *hand*, *mountain*, and *nighttime*.

Run: Stig The *Stig* run is comparable to the Captain Slow run, except for the kernel. For this run we rely on the approximate histogram intersection kernel suggested by Maji *et al.* [11]. This run achieved a mean infAP of 0.0831. As expected this run is outperformed in terms of accuracy by the *Captain Slow* run. For almost all concepts it obtains a slightly lower infAp, with an 8% decrease overall. However, in terms of performance the histogram intersection kernel is much better.

Run: Jezza The *Jezza* run is our cross-domain run. It uses the same implementation as the *Stig* run, but extends

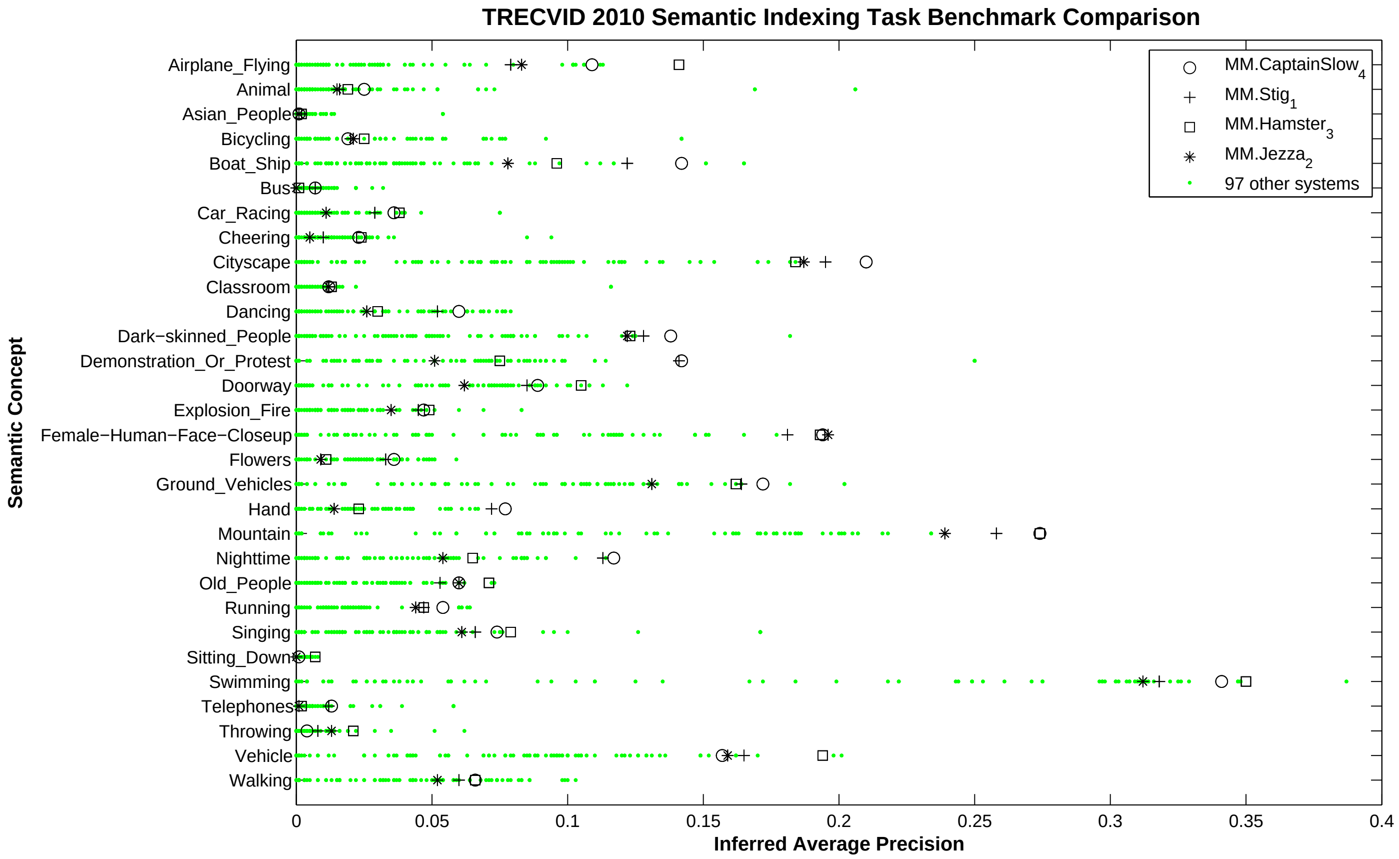

Figure 2: Comparison of MediaMill video concept detection experiments with other concept detection approaches in the TRECVID 2010 Semantic Indexing task.

the number of training samples by sampling additional positive examples from the TRECVID 2005-2009 benchmarks. This run achieved a mean infAP of 0.0685, with the overall highest infAP for 1 concept: *female human face closeup*. For most concepts this run performs slightly worse than the within-domain experiment, but for a few concepts the loss is large.

Run: Hamster The *Hamster* run is our second cross-domain run. It uses the same setting as *Jezza*, except for the multi-frame combination where we replace AVG with MAX. Apparently MAX is a much better choice for the Internet video collection with relatively noisy, motion-full, and short visual content. This run achieved a mean infAP of 0.0830, with the overall highest infAP for 2 concepts: *airplane flying* and *mountain*. It outperforms the *Jezza* run for almost all concepts. In the final version of our notebook paper we will also compare with a within-domain run using a MAX multi-frame combination.

2.5 130 Robust Concept Detectors

We have employed our *Stig* run setting on the entire concept set of TRECVID 2010. All 130 detectors are included in the 2010 MediaMill semantic video search engine for the retrieval experiments.

3 Automatic Video Retrieval

The MediaMill team continued its effort on automatic search in the known item search task, this year submitting two automatic runs. The overall architecture of the search system was based on three different sources of search information — transcripts, detectors, and manually created metadata.

This year we submitted following two automatic search runs for official evaluation. The purpose of these two runs was to investigate the effect of adding search with automatically generated metadata to search with manually created metadata on retrieval performance.

Face: Manual metadata-based search.

BA: Manual metadata-based search + transcript-based search + detector-based search.

We generated four additional (unsubmitted) runs, in order to evaluate both the retrieval performance of individual sources of automatically generated metadata and their combination with manually created metadata.

Transcript Only Transcript-based search.

Detector Only Detector-based search

Detector and Manual Metadata Detector-based search + manual metadata-based search

Transcript and Manual Metadata Transcript-based search + manual metadata-based search

3.1 Retrieval Approaches

All runs were based on the original topic text — the visual cue text was not used, as we found that this text sometimes omitted pertinent (visual) information. Our retrieval approaches were implemented as follows:

Transcript-Based Search and Manual Metadata-Based Search As transcripts and the manual metadata are both text, we used the same approach to search though each of them. We indexed the transcripts and manual metadata for each video using the Indri[1] search engine. The text was normalized, frequently occurring stop words were removed, and we applied Porter stemming. We created three indexes: one with transcripts, one with manual metadata, and one with transcripts and manual metadata together.

At query time, the query was passed to an index and retrieval was done using language modeling approach to retrieval, with the Dirichlet smoothing method [33] with default parameter settings. The *Face* run and the Detector and Manual Metadata runs used the manual metadata index, the Transcript Only run used the transcript index, and the *BA* and Transcript and Manual Metadata runs used the index with transcripts and manual metadata together.

Detector-based search Detector-based search, using our lexicon of 130 robust concept detectors, consisted of two main steps: 1) concept selection and 2) detector combination.

As visual examples were not available for the known item search task, we used text matching to select the appropriate concept detectors to use for a query. Known item search is a high-precision task, and therefore our goal was to only select those detectors that were very closely related to the query. We did this by associating each detector to a list of synonyms, that was then used to match concepts to queries. The synonym lists were created by using WordNet to first identify potential synonyms, and then manually editing the synonym list to remove terms that were expected to cause topic drift, as *track* in the synonym *railroad track* for the concept *railroad*.

Once a set of concept detectors was selected for a topic, the score for each shot in the collection was determined using unweighted averaging of the concept detector scores. The shot-level scores were aggregated to the video level using the passage-based approach described in [5].

Table 1: Automatic search results in terms of mean inverted rank, for submitted and unsubmitted runs.

Face	0.23
BA	0.24
Transcript Only	0.09
Detector Only	0.00
Detector and Manual Metadata	0.23
Transcript and Manual Metadata	0.25

Search Fusion From past TRECVID benchmarks we know that query-dependent fusion approaches have been instrumental in achieving maximum retrieval performance. However, as the known item search task was new this year, we used a query-independent combination approach. Our result analysis will show that there is potential to improve performance by using a query-dependent approach.

Our search fusion approach focused on combining results from detector-based search with results from the various text-based searches. To achieve this we normalized both result lists using Borda normalization. We then combined the two lists using weighted CombSUM fusion [32]. The weighting parameters were derived from a set of training queries; detectors were assigned a weight of 0.15, and the text-based results were assigned a weight of 0.85.

3.2 Automatic Known Item Search Results

An overview of the search results is given in Table 1. We will start by discussing the two official submitted runs, *BA* and *Face*.

Of the two officially submitted runs, the *BA* run gained a slightly higher score than the *Face* run, with a mean inverted rank of 0.24 vs 0.23. At the topic level, *BA* improved over *Face* for 44 topics, and performed worse for 61 topics. Looking at extreme changes in performance at the topic level, for two topics the inverted rank increased from 0 to 1; in other words, the known item went from not being retrieved at all on the basis of manually created metadata, to being ranked as the best result when including automatically generated metadata. These two topics were topic 0241, *Find the video of a beekeeper opening a hive*, and topic 0260, *Find the video of a Mike Moon Production with a man in jacket with orange collar explaining about the Canadian holiday celebrating Queen Victoria's birth.* For each of these topics, the known item did not have any manually created text that matched the topic text, while relevant keywords were mentioned in its ASR transcript multiple times. In addition, appropriate detectors were selected. Topic 0241 was matched to the *outdoor* and *daytime outdoor* detectors, and topic 0260 was matched to the *male person* detectors. For one topic the inverted rank decreased from 1 to 0; the known item was ranked as the best result using manually created metadata, but was not retrieved in the top 100 results when automatically generated metadata was included in the search. This was topic 0031, *Find the narrated video*

[1] http://www.lemurproject.org/indri/

showing Life Quest documents and a box of products. A woman spreads PB&J on a sandwich with a knife. Here the words "Life Quest" were repeated multiple times in the manually created metadata of the known item, but not in its ASR transcript. The *female person* detector was selected for this topic. The results that were returned by the *BA* run for this topic were dominated by non-relevant documents that contained words such as "woman" and "life" in the transcripts and manually created metadata, and that gained a high score for the *female person* detector.

Turning to the unsubmitted runs, we start with the runs based on individual sources of automatically generated metadata, the Transcript Only run and the Detector Only runs. The Detector Only run was exceptional in that it attained a mean performance that approached 0. The Transcript Only run, on the other, gained a mean score that was 39% of the *Face* run (which uses manual metadata only). The Transcript Only run managed to retrieve the known item in the top 100 results for 57 topics. The known item was returned at the top of the result list for 20 topics — in these cases search on transcript data alone was sufficient to identify the known item. These topics include both very general text such as topic 0090, *Find the video of a baby held by mother who gives the baby a bottle of juice*, and topics containing specific named entities such as topic 0094, *Find the video of Mikoyan at Macy's, Wall Street, and UN and bishop.* The Detector Only run retrieved the known item in the top 100 results for 13 topics. In no case did this run place a result at the top of the result list; the topic with highest inverted rank was topic 0137, *Find the video of newsman showing images of Oprah Winfrey, Barack Obama, and Mike Huckabee*, with a score of 0.33. Here the *reporter* detector had been selected for retrieval. This detector placed the known item at the third position in the result list.

Finally, we turn to examine the effect of combining transcripts and detectors with manually created metadata. Looking first at the combination of detectors with manually created metadata in the Detector and Manual Metadata Run, this run improved over the *Face* run (which consists of manual metadata only) for 28 topics. This indicates that detectors can be more useful as a source of reranking information for the known-item search task than as a stand-alone source of search information. However, detectors tend to hurt more often than they help, and for 40 topics inverted rank decreased upon including detectors. This points to a need for query-dependent approaches for including detector-based search results. As for the combination of transcripts with manually created metadata, the Transcript and Manual Metadata run gained the highest mean inverted rank of all runs. At the topic level, the Transcript and Manual Metadata run improved over *Face* for 39 topics, while it performed worse than Face for 61 topics. Including transcripts in search, like detectors, hurts more often than it helps. However, when transcripts help, they help a lot. These results indicate that a query-dependent approach, one that identifies those queries where transcript-based search and detector-based search are useful and assign weights accordingly, is essential for achieving high overall performance in the known item search task.

4 Interactive Video Retrieval

We submitted two interactive runs to the known item search task, *Hannibal* and *Murdock*. Both runs used the same system, but with a different searcher.

4.1 Interactive engine

Our approach for the known item task was to combine metadata search and extensive metadata visualization into one interactive system. For this, we extended previous work on MediaTable [3]. This generic video categorization system allows us to rapidly create specialized interfaces for specific tasks. In particular: in our TRECVID experiments we wanted to see whether the categorization based approach was extendible to known item search.

We chose to combine several metadata and content based resources together in one interface. We added all available metadata as provided by `archive.org`, and we added the 130 robust concept detectors from our lexicon. This information was displayed in one huge table, with each row depicting a single video. The various shots of each video are then depicted using keyframes across multiple columns in the interface. This combination provides a quick overview of each video.

Besides the availability of metadata for search we designed several specialization interfaces specifically for the known item search task:

- A metadata search engine based on Lucene, which allowed us to search through title, description and ASR texts in both English and foreign languages using a rich query language.

- To allow users to rapidly review video content we integrated a video player which shows the currently selected clip at configurable frame rates. This allowed reviewing at definable speeds between 0.5 - 20x the speed of regular video.

- To provide further information about individual videos, we integrated a detail pane. This showed various kinds of information on the selected clip, including title, description, a list of detected semantic concepts, various other available metadata fields, keyframes of all included shots and the automated speech recognition results.

We extended the metadata search pane to also provide results from automatic retrieval of the first 24 topics upon user request. To allow searchers to know that they found the correct item we added validation with the provided known item search Oracle from the detail pane. This system was

Condition	# Topics	# Found
A metadata query only	14	10
B metadata + synonym search	3 more	0
C visual only	5	0
total	22	10

Figure 3: Categorization of topic types of the known item search task. Our MediaTable system incorporates querying using types A and C.

based on manually clicking a button with an option to do this automatically after a set time delay of inspection. If the item was relevant the detail pane would turn green to alert the user that this item was "known".

4.2 Interactive Known Item Search Results

We set up our TRECVID experiments to see whether a categorization based browsing system alone would be sufficient for known item retrieval. We found that our approach yielded a Mean Inverted Rank of 0.45 for *Hannibal* (9 found topics) and 0.41 for *Murdock* (8 found topics)

For a more detailed inspection we split the topics into categories based on their retrievability. First, there are topics of which search for individual words in the query are sufficient to find the required item. Second, there are topics for which searching for synonyms or hypernyms of words in the query allows user to find the result. Third, a minority of topics did not have sufficient metadata annotation at all, and could only be retrieved using visual content analysis. See figure 3 for the categorization.

MediaTable turns out to be most effective for retrieval of topics of type A, where information could be found using metadata only. In these cases, results are typically found in under 30 seconds (data not shown). Preliminary analysis further shows that improvements can be made into the other types of results when we allow interactive combinations of available visual analysis techniques, such as semantic concept detectors.

The lack of synonym/hypernym search did however limit the number of metadata results that could have been found. For a couple of topics (see figure 3) results were not found because searchers were looking for the wrong keyword, e.g. *war* instead of *battle* or *metro* instead of *tube*. Adding integrated synonym/hypernym support to the search engine might have caused the searchers to find these results also.

Lastly, we found that the task setup for known item search is not ideal. The Oracle approach allows users to "see" if this was the item they were looking for. However, humans recognize real known items probably earlier by (visual) recognition of the correct item amongst a larger set of other items. Furthermore, non-relevant results would probably be dismissed faster by the user. In the current setup, these kinds of analysis are difficult to make, and it would be interesting to setup a factual known item search task where the users actively know the item they are looking form.

Although the Oracle does address this issue, we found that deciding when to use the Oracle a tricky decision. To get more into the known item search scenario of the previous paragraph we have considered consulting the Oracle for every video shown the screen, but decided against this, and only consult on explicit user requests.

Overall we conclude that MediaTable allows users to make rapid decisions on metadata-based known item search tasks, allowing users to know whether something is in the collection or not in under 30 seconds. In the current setup we did however not exhaustively look at integration of content based techniques and we feel that further integration would improve results further.

5 Instance Search Task

An instance is a frame from a video and it is used as the query image based on which other frames or videos are retrieved. Our approach uses concept detection methods in order to seek for semantically similar image instances on videos. More specifically, each query is being treated as a different concept which we model based on positive and negative examples. Given each query q similar images are available. These images are used as positive examples for that query concept. For negative examples the image space is sampled and n dissimilar images are kept.

Then, an SVM classifier is trained based on the visual word histograms of these positive and negative examples. We follow the same pipeline explained in Section 2, but we limit the number of visual descriptors to SIFT and RGB-SIFT, and we employ soft instead of hard-assignment. The resulting classifier ranks the shots from the test collection, which we use as our retrieval result. The desired scenario would be to select as negative examples images that are close in the visual word feature space, yet not semantically similar. However, random selection works sufficiently well and this is the approach followed in the current setup.

5.1 Instance Search Results

We submitted four runs, all based on random selection of 50 random negative examples. As expected the random selection has a small effect on the overall performance of our approach, all runs obtain a mean infAP of 0.011 or 0.010. Results for individual queries may vary a bit more, but not seriously so. Our simple retrieval method performs best for 3 queries: *zebra stripes on pedestrian crossing*, *interior of Dutch parliament*, and *tank*. Our current approach is less suited for person and character queries, but is competitive for object and, especially, location queries.

6 Lessons Learned

TRECVID continues to be a rewarding experience in gaining insight in the difficult problem of concept-based video

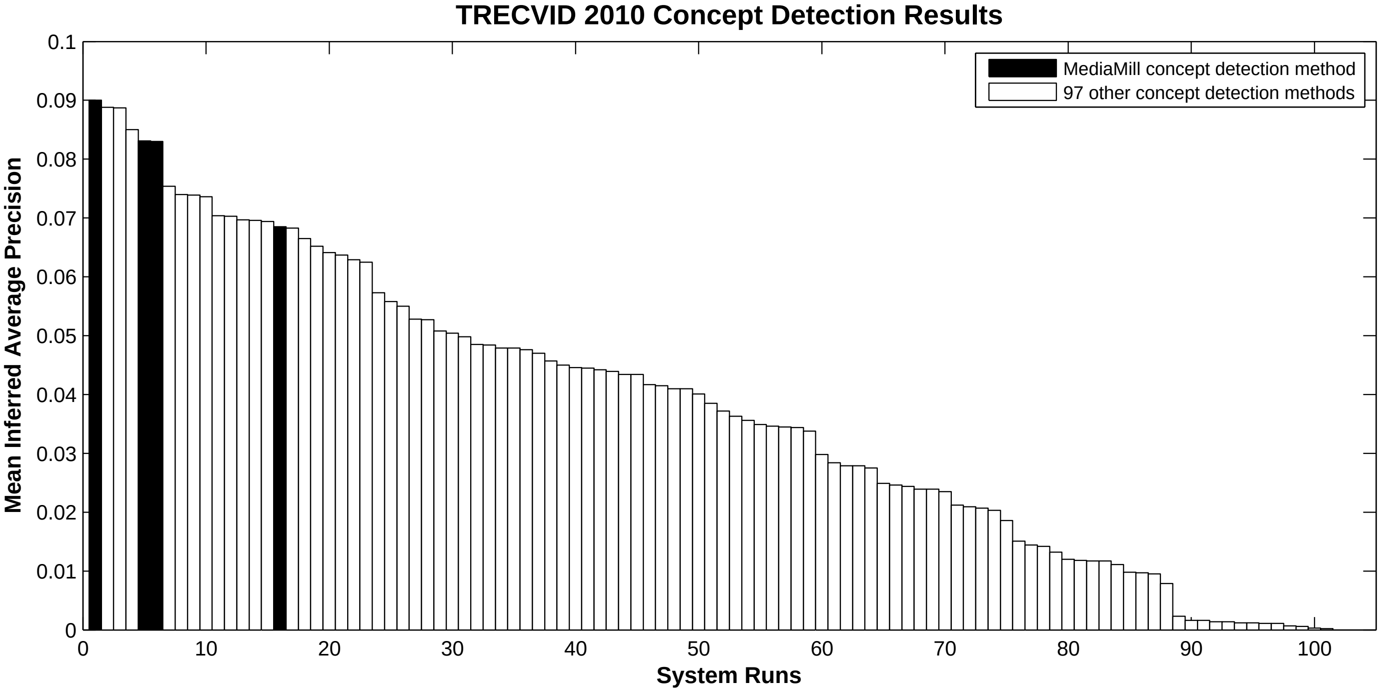

Figure 4: Overview of the 2010 TRECVID concept detection task benchmark in which MediaMill was the best overall performer, all runs ranked according to mean inferred average precision.

retrieval [17]. The 2010 edition has again been a very successful participation for the MediaMill team resulting in top ranking for concept detection, see Figure 4. In the final version of this manuscript we will highlight our most important lessons learned to conclude the paper.

Acknowledgments

This research is sponsored by the STW SEARCHER project, the BeeldCanon project, and a Fulbright scholarship. The authors are grateful to NIST and the TRECVID coordinators for the benchmark organization effort.

References

[1] G. J. Burghouts and J.-M. Geusebroek. Performance evaluation of local color invariants. *Computer Vision and Image Understanding*, 113(1):48–62, 2009.

[2] 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/`.

[3] O. de Rooij, M. Worring, and J. J. van Wijk. Mediatable: Interactive categorization of multimedia collections. *IEEE Computer Graphics and Applications*, 30(5):42–51, 2010.

[4] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scence categories. pages 524–531, 2005.

[5] B. Huurnink, C. G. M. Snoek, M. de Rijke, and A. W. M. Smeulders. Todays and tomorrows retrieval practice in the audiovisual archive. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, pages 18–25, Xian, China, July 2010. Best paper runner-up.

[6] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *IEEE International Conference on Computer Vision*, pages 604–610, 2005.

[7] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178, New York, USA, 2006.

[8] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *Int'l J. Computer Vision*, 43(1):29–44, 2001.

[9] H.-T. Lin, C.-J. Lin, and R. C. Weng. A note on Platt's probabilistic outputs for support vector machines. *Machine Learning*, 68(3):267–276, 2007.

[10] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.

[11] S. Maji, A. C. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 619–626, Anchorage, Alaska, 2008.

[12] M. Marszałek, C. Schmid, H. Harzallah, and J. van de Weijer. Learning object representations for visual object class recognition, October 2007. Visual Recognition Challange workshop, in conjunction with ICCV.

[13] C. Petersohn. Fraunhofer HHI at TRECVID 2004: Shot boundary detection system. In *Proceedings of the TRECVID Workshop*, NIST Special Publication, Gaithersburg, USA, 2004.

[14] J. C. Platt. Probabilities for SV machines. In A. J. Smola, P. L. Bartlett, B. Schölkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 61–74. The MIT Press, Cambridge, USA, 2000.

[15] Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.

[16] J. Sivic and A. Zisserman. Efficient visual search for objects in videos. *Proceedings of the IEEE*, 96(4):548–566, 2008.

[17] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVid. In *Proceedings of the ACM SIGMM International Workshop on Multimedia Information Retrieval*, pages 321–330, 2006.

[18] C. G. M. Snoek and A. W. M. Smeulders. Visual-concept search solved? *IEEE Computer*, 43(6):76–78, June 2010.

[19] C. G. M. Snoek, K. E. A. van de Sande, O. de Rooij, B. Huurnink, J. R. R. Uijlings, M. van Liempt, M. Bugalho, I. Trancoso, F. Yan, M. A. Tahir, K. Mikolajczyk, J. Kittler, M. de Rijke, J.-M. Geusebroek, T. Gevers, M. Worring, D. C. Koelma, and A. W. M. Smeulders. The MediaMill TRECVID 2009 semantic video search engine. In *Proceedings of the 7th TRECVID Workshop*, Gaithersburg, USA, November 2009.

[20] C. G. M. Snoek, K. E. A. van de Sande, O. de Rooij, B. Huurnink, J. C. van Gemert, J. R. R. Uijlings, J. He, X. Li, I. Everts, V. Nedović, M. van Liempt, R. van Balen, F. Yan, M. A. Tahir, K. Mikolajczyk, J. Kittler, M. de Rijke, J.-M. Geusebroek, T. Gevers, M. Worring, A. W. M. Smeulders, and D. C. Koelma. The MediaMill TRECVID 2008 semantic video search engine. In *Proceedings of the 6th TRECVID Workshop*, Gaithersburg, USA, November 2008.

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

[22] C. G. M. Snoek, M. Worring, J.-M. Geusebroek, D. C. Koelma, and F. J. Seinstra. On the surplus value of semantic video analysis beyond the key frame. In *Proceedings of the IEEE International Conference on Multimedia & Expo*, Amsterdam, The Netherlands, July 2005.

[23] C. G. M. Snoek, M. Worring, J.-M. Geusebroek, D. C. Koelma, F. J. Seinstra, and A. W. M. Smeulders. The semantic pathfinder: Using an authoring metaphor for generic multimedia indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1678–1689, October 2006.

[24] T. Tuytelaars and K. Mikolajczyk. Local invariant feature detectors: A survey. *Foundations and Trends in Computer Graphics and Vision*, 3(3):177–280, 2008.

[25] J. R. R. Uijlings, A. W. M. Smeulders, and R. J. H. Scha. Real-time visual concept classification. *IEEE Transactions on Multimedia*, 12(7):665–681, 2010.

[26] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, September 2010.

[27] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Empowering visual categorization with the GPU. *IEEE Transactions on Multimedia*, 2011. Submitted.

[28] J. C. van Gemert, C. G. M. Snoek, C. J. Veenman, A. W. M. Smeulders, and J.-M. Geusebroek. Comparing compact codebooks for visual categorization. *Computer Vision and Image Understanding*, 114(4):450–462, April 2010.

[29] J. C. van Gemert, C. J. Veenman, and J. M. Geusebroek. Episode-constrained cross-validation in video concept retrieval. *IEEE Transactions on Multimedia*, 11(4):780–785, 2009.

[30] J. C. van Gemert, C. J. Veenman, A. W. M. Smeulders, and J. M. Geusebroek. Visual word ambiguity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(7):1271–1283, 2010.

[31] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, USA, 2nd edition, 2000.

[32] P. Wilkins. *An investigation into weighted data fusion for content-based multimedia information retrieval*. PhD thesis, Dublin City University, Dublin, Ireland, 2009.

[33] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems*, 22(2):179–214, 2004.

[34] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *International Journal of Computer Vision*, 73(2):213–238, 2007.