The MediaMill TRECVID 2004 Semantic Video Search Engine

C.G.M. Snoek, M. Worring, J.M. Geusebroek, D.C. Koelma, and F.J. Seinstra
MediaMill, University of Amsterdam
Kruislaan 403, 1098 SJ Amsterdam, The Netherlands
{cgmsnoek, worring, mark, koelma, fjseins}@science.uva.nl

Abstract

This year the UvA-MediaMill team participated in the Feature Extraction and Search Task. We developed a generic approach for semantic concept classification using the semantic value chain. The semantic value chain extracts concepts from video documents based on three consecutive analysis links, named the content link, the style link, and the context link. Various experiments within the analysis links were performed, showing amongst others the merit of processing beyond key frames, the value of style elements, and the importance of learning semantic context. For all experiments a lexicon of 32 concepts was exploited, 10 of which are part of the Feature Extraction Task. Top three system-based ranking in 8 out of the 10 benchmark concepts indicates that our approach is very promising. Apart from this, the lexicon of 32 concepts proved very useful in an interactive search scenario with our semantic video search engine, where we obtained the highest mean average precision of all participants.

1 Introduction

Technological developments in a wide range of disciplines are facilitating the access to large multimedia repositories at a semantic level. Sophisticated detection methods for specific semantic concepts exist. However, because the enormous amount of possible concepts in video documents, research should concentrate on generic methods for concept detection, see for example [1, 5, 19].

Although substantial progress has been achieved, the semantic gap still hampers commercial exploitation of concept detection methods. In [16], similarity, learning, and interaction are identified as key techniques that aid in bringing semantics to the user. Thus, given the semantic gap, an ideal semantic video retrieval system should be able to learn a large set of concepts for initial search, and use similarity and interaction to refine results to an acceptable level.

In this contribution we propose the semantic value chain, a novel method for generic semantic concept detection. We developed detectors for a lexicon of 32 concepts that allow for query by concept. Furthermore, we explored the combination of query by concept, query by similarity, and interaction into an integrated semantic video search engine. To demonstrate the effectiveness of our approach, both the semantic value chain and interactive search experiments are evaluated within the 2004 TRECVID video retrieval benchmark.

The organization of this paper is as follows. First, we discuss the semantic lexicon used in our system. Then we proceed in Section 3 with the description of the semantic value chain. In Section 4 we elaborate on our semantic video search engine. Benchmark results are discussed in Section 5.

2 Semantic Lexicon

A priori we define a lexicon of 32 semantic concepts. Concepts are chosen based on the indices described in [18], previous TRECVID feature extraction tasks, anticipated positive influence on the result of the 10 benchmark concepts, as well as being relevant for general search. For all concepts

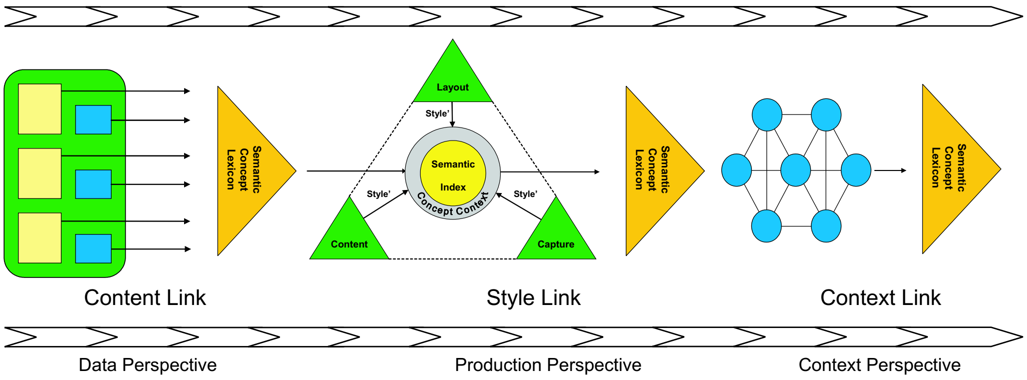


Figure 1: *The semantic value chain.*

considered a ground truth is annotated. The ground truth is based on a cleaned version of the common annotation effort of the TRECVID 2003 development set [9], the evaluation results of the 2003 test set provided by NIST, and additional annotations of the TRECVID 2004 development set. All our submitted runs can thus be considered to be of type B. The following concepts form the semantic lexicon:

- {*airplane take off, American football, animal, baseball, basket scored, beach, bicycle, Bill Clinton, boat, building, car, cartoon, financial news anchor, golf, graphics, ice hockey, Madeleine Albright, news anchor, news subject monologue, outdoor, overlayed text, people, people walking, physical violence, road, soccer, sporting event, stock quotes, studio setting, train, vegetation, weather news*};

Together with the video data, the annotated lexicon forms the input for the semantic value chain.

3 Semantic Value Chain Analysis

For each semantic concept in the lexicon a tailored approach could be developed, however we strive for a generic method. To arrive at such a generic approach for concept detection in video, we view semantic video analysis as an inverted authoring process [18]. To express a semantic intention an author uses style elements. In [19] we identified four style elements, namely: layout, content, capture, and concept context, that aid in generic extraction of semantics from produced video. It was found that additional analysis methods for content and concept context have the largest potential to improve semantic index results. In this contribution we therefore propose the semantic value chain. The output of each link in the chain forms the input for the next link, in the process enriching the semantics. The semantic value chain extracts concepts from video based on three analysis links, i.e. the content link, the style link, and the context link. In this Section we first discuss the general architecture used in each link. Then we proceed with the individual analysis links for content, style, and context. A complete overview of the semantic value chain is given in Fig. 1.

3.1 General Link Architecture

We view detection of concepts in video as a pattern recognition problem, where the aim is to detect a semantic concept ω based on a pattern x. To obtain x a granularity needs to be chosen first, e.g. a camera shot segmentation. Each link in the semantic value chain has a separate analysis method to obtain x from a video, a shot segmentation, and an annotated lexicon. To learn the relation between ω and x we exploit supervised learning by means of statistical pattern recognition.

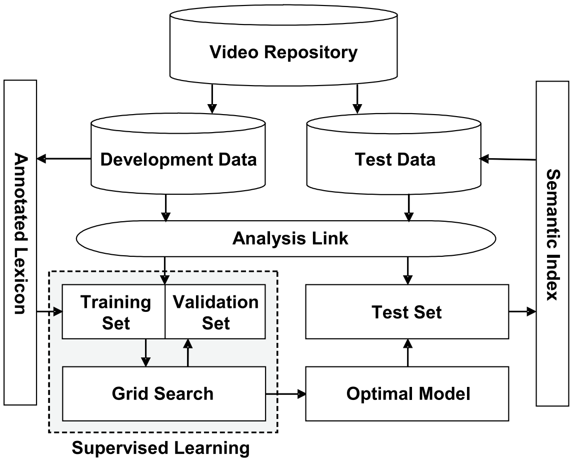


Figure 2: *General link architecture within the semantic value chain.*

Among the large variety of supervised machine learning approaches available, the Support Vector Machine (SVM) framework [20, 4] has proven to be a solid choice [17, 1]. The SVM is able to learn from few examples, handle unbalanced data, and handle unknown or erroneous detected data. An SVM tries to find an optimal separating hyperplane between two classes by maximizing the margin between those two different classes. Finding this optimal hyperplane is viewed as the solution of a quadratic programming problem. We convert the SVM margin to a posterior probability using Platt's method [11]. Hence, probabilistic models, obtained when an SVM is trained for a semantic concept ω, result in a likelihood $p(\omega|x)$ when applied to unseen patterns from the test data.

The influence of SVM parameters on concept detection performance is significant [10]. To obtain optimal parameter settings for a semantic classifier, grid search on a large number of classifier parameter combinations must be applied by using an independent validation set. A priori we therefore split the TRECVID 2004 development data into a non-overlapping training and validation set. The training set $\mathcal{D}$ contained 85% of the development data, the validation set $\mathcal{V}$ contained the remaining 15%. Based on the broadcast date a proportional number of videos are alternatingly assigned to each set. Apart from the amount of data, this division assures maximum comparability for both sets. The predefined training and validation set are used in combination with 3-fold cross validation to optimize concept detection performance.

Each analysis link in the semantic value chain exploits feature extraction to obtain pattern x from the data. Then, it uses a supervised learning module to learn an optimal model for all concepts in the lexicon. This is illustrated in the overview of our general link architecture in Fig. 2.

3.2 Content Link

In the content link we view of video from the data perspective. In general, three data streams exist in video, i.e. the auditory, textual, and visual modality. For this years benchmark the content link exploits text and visual features.

3.2.1 Visual Analysis

We analyze the visual modality at the image level. First, we remove the border of each frame, including the space occupied by a possible ticker tape. Then, we analyze 1 out of every 15 frames to limit the dependency of chosen key frames. In those frames, we aim for weak segmentation, i.e. a segmentation of an image into internally homogenous regions based on some set of visual feature detectors [16]. Invariance was identified in [16] as a crucial aspect of a visual feature detector, e.g.

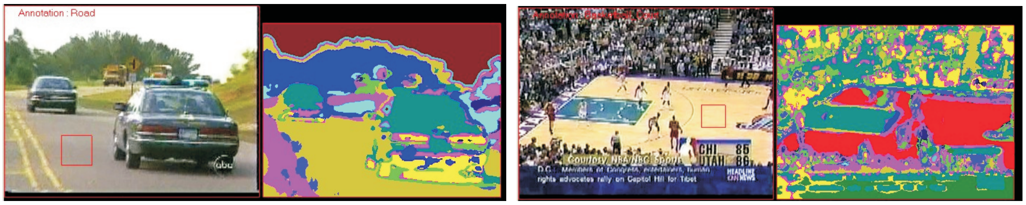


Figure 3: *Examples of regional visual concept segmentation.*

to design features which limit the influence of accidental recording circumstances. As the conditions under which semantic concepts appear in large video repositories may vary greatly, we use invariant visual features to arrive at weak segmentation. More specifically, visual features extracted by using Gaussian color invariant measurements [7].

To obtain the visual features, we decorrelate RGB color values by linear transformation to the opponent color system [7]. Smoothing the values with a Gaussian filter suppresses acquisition and compression noise. The size of the Gaussian filters is varied to obtain a color representation that is compatible with variations in the target object size. Normalizing each opponent color value by its intensity suppresses global and local intensity variations. This results in two chromaticity values per color pixel. Furthermore, we obtain rotationally invariant features by taking Gaussian derivative filters, and combining the responses into two chromatic gradients. The seven measurements in total, and each calculated over three scales, yield a 21 dimensional feature vector per pixel. This vector serves as the input for a multi-class SVM [4] that associates each pixel to one of the following regional visual concepts:

- {*colored clothing, concrete, fire, graphic blue, graphic purple, graphic yellow, grassland, greenery, indoor sport court, red carpet, sand, skin, sky, smoke, snow/ice, tuxedo, water body, wood*};

As our visual feature analysis method is based on invariance we only need a few examples, in practice less then 10 per class are sufficient. This pixel-wise classification results in a weak segmentation of an image frame in terms of regional visual concepts, see Fig. 3 for an example.

Segmenting image frames into regional visual concepts at the granularity of a pixel is computationally intensive. Especially, if you aim to analyze as many frames as possible. Hence, we have to solve a performance problem. For the processing of the visual modality in the content link we have therefore applied the Parallel-Horus software architecture [15]. This architecture, consisting of a large collection of low-level image processing primitives, allows the programmer to write *fully sequential* applications for efficient parallel execution on homogeneous clusters of machines. While we estimate that the processing of the entire TRECVID data set would have taken over 250 days on the fastest sequential machine available to us, application of Parallel-Horus in combination with a distributed Beowulf cluster consisting of 200 dual 1-Ghz Pentium-III CPUs [3] reduced the processing time to less than 48 hours [15].

After segmentation of every 15th frame, the percentage of pixels associated to each of the 18 regional visual concepts is used as an image feature vector $\vec{i}$. This vector forms the input for an SVM that associates a probability $p(\omega|\vec{i})$ to each frame for all 32 classes in the general lexicon of concepts. We use a combination of classification results for individual frames over time to generate a probability at shot level. For this purpose we evaluated four combination functions, namely: minimum, maximum, average, and product. To optimize parameter settings, we use 3-fold cross validation on $\mathcal{D}$. We then test the obtained optimal model for each combination function on $\mathcal{V}$. We found that averaging the results of single images within a shot results in much better performance in terms of average precision than an approach that relies on one key frame only.

Table 1: *Semantic concept ordering based on content link analysis performance.*

1.	Weather news	9.	People walking	17.	Golf	25.	Road
2.	Stock quotes	10.	Financial anchor	18.	People	26.	Beach
3.	Anchor	11.	Ice hockey	19.	American football	27.	Train
4.	Overlayed text	12.	Cartoon	20.	Outdoor	28.	Madeleine Albright
5.	Basket scored	13.	Studio setting	21.	Car	29.	Building
6.	Graphics	14.	Physical violence	22.	Bill Clinton	30.	Airplane take off
7.	Baseball	15.	Vegetation	23.	News subject monologue	31.	Bicycle
8.	Sporting event	16.	Boat	24.	Animal	32.	Soccer

3.2.2 Speech-based Textual Analysis

Transcribed speech obtained by the LIMSI speech detection system [6] serves as the textual input. After stopword removal using SMART's English stoplist [12], text that falls within the boundaries of a camera shot is associated to that shot[1]. Since a semantic concept is also associated with a shot, we learn a lexicon of words that have an association to a concept.

We compare the text associated with each shot with the learned lexicon to construct a text vector $\vec{t}$. This vector contains the frequency histogram of words that have an association to a concept. Because we treat all words in the lexicon equally, $\vec{t}$ contains responses to words that are likely to be related to a semantic concept, but also words that have no obvious relation to a concept. For the concept *train* for example the lexicon contains logically related words like passenger, tracks, train, locomotive, overpass, and freight, but also less likely related words, e.g. cars, world, today, and twelve. To prevent the influence of domain knowledge we apply an SVM on $\vec{t}$ to learn which combination of words is important for a certain semantic concept. The SVM assigns a probability $p(\omega|\vec{t})$ to each shot, for all concepts in the lexicon. We use 3-fold cross validation on $\mathcal{D}$ to optimize parameter settings for the learned models.

3.2.3 Submitted Runs from the Content Link

Based on the sketched analysis methods we submitted two runs from the content link. Results are submitted for a subset of 10 semantic concepts from the lexicon, that is evaluated within the benchmark.

The first run is based on the best unimodal performance of a semantic concept on $\mathcal{V}$ (BU). Except for *basket scored*, all semantic concepts had better performance for speech based text analysis than for visual analysis. The second run uses vector fusion (VF) to integrate visual and speech based text analysis. Both vectors $\vec{i}$ and $\vec{t}$ serve as input for this integrated analysis method. We concatenate the text vector $\vec{t}$ with one image vector from each camera shot that has maximum probability for a semantic concept, $\vec{i}_{max}$, into an integrated multimodal vector $\vec{m}$. This vector serves as the input for an SVM that associates probability $p(\omega|\vec{m})$ to each shot, for all 32 concepts in the lexicon. Again we use 3-fold cross validation on $\mathcal{D}$ for parameter optimization.

The VF run forms the input for the next link in the semantic value chain. For all concepts we compute the average precision performance on $\mathcal{V}$. An overview of all concepts ranked according to average precision validation performance in the content link is given in Table 1.

[1] For Person X related concepts we stretch the camera shot boundaries with five seconds on each side, as in broadcast news names or other indicative words are often mentioned just before or after a person is visible.

3.3 Style Link

In the style link we view a video from the production perspective. Based on the methodology presented in [19], this link analyzes a produced video based on a set of four style detectors related to layout, content, capture, and concept context. We combine style detector results into an iterative classifier combination scheme to extract the semantics.

3.3.1 Style Detectors

We develop detectors for all four style roles as feature extraction in the style link, see [19] for specific implementation details. We have chosen to categorize the output of all style detectors, as this allows for easy fusion.

For the layout $\mathcal{L}$ the length of a camera shot is used as a feature, as this is known to be an informative descriptor for genre [18]. Overlayed text is another informative descriptor. Its presence is detected by a text localization algorithm [13]. To segment the auditory layout, periods of speech and silence are detected based on an automatic speech recognition system [6]. We obtain a voice over detector by combining the speech segmentation with the camera shot segmentation [19]. The set of layout features is thus given by: $\mathcal{L} = \{$*shot length, overlayed text, silence, voice over*$\}$.

As concerns the content $\mathcal{C}$, a frontal face detector [14] is applied to detect people. We count the number of faces, and for each face its location is derived [19]. Apart from faces, we also apply a car detector [14] to check for presence of cars. In addition, we measure the average amount of object motion in a camera shot [17]. Based on speaker identification [6] we have been able to identify each of the three most frequent speakers. The camera shot is checked for the presence on the basis of speech from one of the three [19]. Text strings recognized by Video Optical Character Recognition [13] are checked on length [19]. They are used as input for a named entity recognizer [21]. On the transcribed text obtained by the LIMSI automatic speech recognition system, we also apply named entity recognition. The set of content features is thus given by: $\mathcal{C}$ =$\{$*faces, face location, cars, object motion, frequent speaker, overlayed text length, video text named entity, voice named entity*$\}$.

For capture $\mathcal{T}$, we compute the camera distance from the size of detected faces [14, 19]. In addition to camera distance, several types of camera work are detected [2]. Finally, for capture we also estimate the amount of camera motion [2]. The set of capture features is thus given by: $\mathcal{T} = \{$*camera distance, camera work, camera motion*$\}$.

Concept context allows to enhance or reduce correlation between semantic concepts. For the initial concept context $\mathcal{S}$ we developed a reporter detector. Reporters were recognized by fuzzy matching of strings obtained from the transcript and VOCR with a database of names of CNN and ABC affiliates [19]. The semantic results of the content link serve as the most important detector for the concept context. Based on the order defined in Table 1 a concept detection result is iteratively added to the concept context. Results for all concepts are ranked according to the maximum obtained probability in the content link, i.e. $p(\omega|\vec{m})$ or $p(\omega|\vec{t})$. We use this rank to assign a semantic concept detector result into one of five categories. The basic set of concept context detectors is given by: $\hat{\mathcal{S}} = \{$*reporter, content link rank*$\}$.

The concatenation of $\left\{\hat{\mathcal{L}}, \hat{\mathcal{C}}, \hat{\mathcal{T}}, \hat{\mathcal{S}}\right\}$ yields a style vector $\vec{s}$. This vector forms the input for an iterative classifier that trains a style model for each concept in the lexicon.

3.3.2 Iterative Enrichment

In the concept context, we have defined order and started with *weather news*. This yields our first style vector $\vec{s}_1$. Order is then exploited as follows. We train a style model for the concept *weather news*, ω_1, using $\vec{s}_1$. Based on $p(\omega_1|\vec{s}_1)$, the concept *weather news* is then again added to the concept context or not. The decision to add a concept to the concept context depends on the general threshold τ on $p(\omega|\vec{s})$. In this iterative process the content link rank feature is replaced for the detected concept. Together with the content link rank of semantic concept 2, i.e. *stock quotes*, this yields $\vec{s}_2$. This iterative process is then repeated for all semantic concepts in the lexicon. To optimize parameter settings for all individual style models, we used 3-fold cross validation on $\mathcal{D}$.

3.3.3 Submitted Runs from the Style Link

We perform different experiments to verify the influence of the order. Furthermore we experiment with different values for τ, and check the influence of separate style models for ABC and CNN. The rationale here is that different authors have different style, and this should have an impact on semantic concept performance [19].

In the AC1 run we use the order of Table 1 as a basis. Separate style models for ABC and CNN were created, both using a threshold value of 0.5 for τ In the AC2 run we first train style models for the 22 concepts that were not part of the TRECVID 2004 evaluation. The 10 concepts defined in the feature extraction task are performed at the end. The relative order is again based on the order of Table 1. Like AC1, separate style models for ABC and CNN are created, both using a threshold value of 0.5 for τ The AC3 run is similar to the AC1 run, but τ was now set to 0.1. The COM run combines ABC and CNN and uses the same settings as the AC1 run.

3.4 Context Link

In the context link we view a video from the context perspective. In the context link we rely on concept detectors only. To combine concept detection results, different context configurations can be exploited. We explore two configurations, one based on context vectors, and one based on an ontology.

3.4.1 Context Vectors

Both the content link and style link yield for each concept in the lexicon a probability that the concept is present in a shot. We fuse those probabilities into a context vector $\vec{c}$ for each shot. This vector then serves as the input for a stacked classifier that learns new concepts not present in $\vec{c}$, or tries to improve performance of existing semantic concepts, already present in $\vec{c}$, see also [1, 8]. For TRECVID we only experiment with 32 dimensional context vectors, that aim to improve performance of concepts already in the lexicon. To optimize parameter settings, we use 3-fold cross validation on $\mathcal{V}$.

3.4.2 Ontology

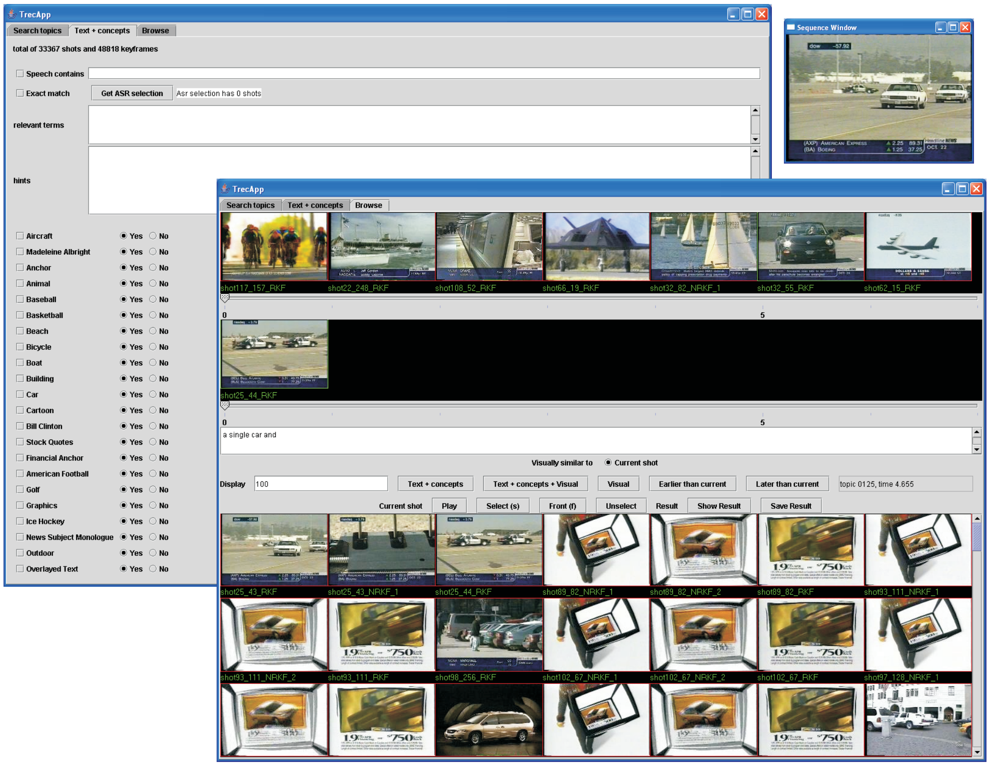
We also experiment with an ontology as an instance of the context configuration. In [23] an ontology based learning algorithm was proposed to improve concept detection results. We use the proposed *confusion factor* to improve results. In short, the method updates probability scores by taking into account that certain concept combinations are very unlikely to co-occur, e.g. *studio setting* and *outdoor*. We define the concept combinations on a set of common sense rules.

3.4.3 Submitted Runs from the Context Link

Based on the above configurations we perform different experiments in the context link. In the CC run we combine the results of the content link into one context vector. In the R4 run we combined the results of the AC1 run into a context vector. In the R5 run we combine the results of the AC2 run into a context vector. Finally, in the OR5 run we experiment with the ontology.

4 Semantic Video Search Engine

For the interactive Search Task we developed a semantic video search engine, which elaborates on last years system [22]. The set of 32 semantic concepts forms the main input for our semantic video search engine and allows for query by concept. Concepts can be queried based on likely presence or absence and by combining results from the two different runs. Apart from query by concept, we also provide



total of 33367 shots and 48818 keyframes

□ Speech contains

□ Exact match — Asr selection has 0 shots

relevant terms

hints

□ Aircraft	● Yes	○ No
□ Madeleine Albright	● Yes	○ No
□ Anchor	● Yes	○ No
□ Animal	● Yes	○ No
□ Baseball	● Yes	○ No
□ Basketball	● Yes	○ No
□ Beach	● Yes	○ No
□ Bicycle	● Yes	○ No
□ Boat	● Yes	○ No
□ Building	● Yes	○ No
□ Car	● Yes	○ No
□ Cartoon	● Yes	○ No
□ Bill Clinton	● Yes	○ No
□ Stock Quotes	● Yes	○ No
□ Financial Anchor	● Yes	○ No
□ American Football	● Yes	○ No
□ Golf	● Yes	○ No
□ Graphics	● Yes	○ No
□ Ice Hockey	● Yes	○ No
□ News Subject Monologue	● Yes	○ No
□ Outdoor	● Yes	○ No
□ Overlayed Text	● Yes	○ No

a single car and

Visually similar to ● Current shot

Display 100

topic 0125, time 4.655

Figure 4: *The MediaMill semantic video search engine. The system allows for interactive query by concept, query by keyword, and query by example. The top op the right panel shows the selected results, the bottom shows results for the semantic concept car.*

users with the possibility for query by keyword. To that end we first derive words from the speech recognition result [6]. Latent Semantic Indexing is then used to reduce the search space, this space is then also used for querying [22]. An exact match of keywords on the transcribed speech is also possible. Finally, our system allows for query by example. For all key frames in the video repository, we compute the global *Lab* color histograms using 32 bins for each channel. The Euclidean distance is used for histogram comparison.

Combining query interfaces allows for interactive retrieval. For search topics that have a close relation to one or more concepts from the lexicon, query by concept can be used. Examples include finding shots of ice hockey rinks, bicycles rolling along, and Bill Clinton in front of an American flag. Query by concept can be combined with query by keyword to find specific instances of semantic concepts, e.g. shots containing flooded buildings, horses in motion, or people walking together with dogs. Of course it can also be used in isolation to find very specific topics, like shots containing Benjamin Netanyahu or Boris Yeltsin. Based on query by example the retrieved results can be augmented with visually similar camera shots. To give an example, we performed an interactive query on the semantic concepts *airplane take off, bicycle, boat, car,* and *train* to detect a set of *vehicles* in Fig. 4.

5 Results

To evaluate both the semantic value chain and our semantic video search engine we participated in the Feature Extraction Task and the Search Task of TRECVID 2004. We will first discuss our results on the Feature Extraction Task.

Table 2: *UvA-MM TRECVID 2004 run comparison for all 10 benchmark concepts. In parentheses the total number of correctly judged semantic concepts.*

	Content Link		*Style Link*				*Context Link*			
	BU	VF	AC1	AC2	AC3	COM	CC	R4	R5	OR5
Boat (441)	0.108	**0.117**	0.096	0.094	0.098	0.101	0.084	0.070	0.096	0.042
Madeleine Albright (19)	**0.238**	0.136	0.023	0.027	0.021	0.035	0.000	0.015	0.018	0.021
Bill Clinton (409)	0.123	0.130	0.150	0.156	0.154	**0.160**	0.135	0.149	0.155	0.105
Train (43)	**0.083**	0.054	0.062	0.050	0.074	0.072	0.041	0.005	0.004	0.023
Beach (374)	0.008	0.020	0.017	0.011	0.010	**0.021**	0.010	0.012	0.010	0.006
Basket scored (103)	0.017	0.118	0.180	**0.214**	0.200	0.141	0.174	0.209	0.193	0.194
Airplane take off (62)	0.051	0.065	0.037	0.042	**0.073**	0.043	0.052	0.040	0.050	0.037
People walking (1695)	0.134	0.150	0.159	0.151	0.138	0.166	0.139	**0.170**	0.168	0.106
Physical violence (292)	0.062	0.064	0.071	0.052	0.076	0.067	0.080	**0.086**	0.069	0.064
Road (938)	0.073	0.089	0.129	0.135	0.135	0.118	0.080	0.138	**0.141**	0.120
MAP	0.090	0.094	0.093	0.095	**0.096**	0.093	0.078	0.089	0.090	0.072

5.1 Semantic Concept Detection

In total ten runs were submitted for the Feature Extraction task, mean average precision (MAP) results are visualized in Table 2, an overview of the precision at 100 is given in Table 3.

From the submitted runs several conclusions can be drawn. First, the combination of modalities (VF) in the content link almost always outperforms unimodal approaches (BU), except for sparse concepts like *Madeleine Albright* and *train*. Second, a simple combination of textual and visual content (VF) yields comparative MAP to more advanced methods, e.g. AC2 and R5. However, for non-sparse concepts, the style link and context link in general improve concept detection performance. Moreover, the average number of hits in the first 100 results increases in each link. This suggests that our method requires a minimal number of examples to be effective. Third, the influence of concept order in the style link seems to be important for the first added concepts only. *Basket scored* for example, benefits a lot from added concept context in the AC2 run, when compared to the AC1 run. However, in terms of MAP the AC2 run is only a little bit better then AC1. Fourth, lowering the value of threshold τ has a positive influence on performance, and results in our best run when measured in terms of MAP. Fifth, although the COM and AC1 run obtain similar MAP, it can be concluded that the SVM classifier architecture is smart enough to make the distinction between ABC and CNN styles. It seems that the SVM profits more from the number of examples than a strict separation in style models. However, for concepts that are only common in one style, e.g. *basket scored* in CNN, results do profit from a distinction between broadcast stations. Sixth, the influence of style is evident. When comparing the CC run with the R4 run, the increase in MAP resulting from the style link is 14%. Finally, the difference in performance between the ontology run and the R5 run is significant (R5 performs 25% better). This shows that making a priori assumptions about semantics in video repositories is not a good idea, its better to learn semantic context from the data. For the concept *boat* for example, we defined a priori that *vegetation* was not very likely to co-occur. This had a very negative influence on performance, because it turned out that one of the boats we found very frequently in the other runs was a kayak in a forest. Ontologies do seem to help for sparse concepts, but this can also be explained as a failure of the learning approach because of a lack of examples.

If we compare our results with all other submitted systems, it is clear that we have presented a powerful generic approach for semantic video indexing. The semantic value chain performs the best for two concepts, second for five concepts, and third for one concept, see Fig. 5.

Table 3: *UvA-MM TRECVID 2004 precision at 100 comparison for all 10 benchmark concepts. In parentheses the total number of correctly judged semantic concepts.*

	Content Link		*Style Link*				*Context Link*				
	BU	VF	AC1	AC2	AC3	COM	CC	R4	R5	OR5	*Average*
Boat (441)	**44**	42	38	36	40	39	34	37	39	37	38.6
Madeleine Albright (19)	10	**12**	5	6	5	5	0	4	4	5	5.6
Bill Clinton (409)	25	26	35	32	34	36	26	37	**41**	33	32.5
Train (43)	**9**	7	7	6	8	8	7	3	2	5	6.2
Beach (374)	10	13	12	11	9	**14**	9	12	11	9	11.0
Basket scored (103)	8	24	21	**35**	30	21	26	30	33	33	26.1
Airplane take off (62)	9	10	8	**11**	9	9	10	8	10	10	9.4
People walking (1695)	65	65	72	57	65	71	68	**83**	77	54	67.7
Physical violence (292)	17	17	25	18	13	24	17	**31**	23	19	20.4
Road (938)	47	43	53	51	53	41	41	51	**55**	52	48.7
Average	24.4	25.9	27.6	26.3	26.6	26.8	23.8	**29.6**	29.5	25.7	26.6

5.2 Interactive Search

For the interactive search we submitted four runs in total. All runs were performed by expert users. One user had knowledge about the semantic concepts and their validation set performance. The others were confronted with the semantic concepts the first time they were introduced to the system.

For all topics the best UvA-MM result of each run is visualized, together with the median and the best overall result, in Fig. 6. We scored above the median for all search topics, had best performance for seven topics, and obtained best overall MAP with run UvA-MM1 (0.352). This run was completed by the user with knowledge about the semantic concepts. The other three users obtained an MAP of 0.227 on average. A score that is 25% higher than the median of all interactive search runs submitted.

We explain the success of our approach, in part, by the lexicon we used for our semantic video

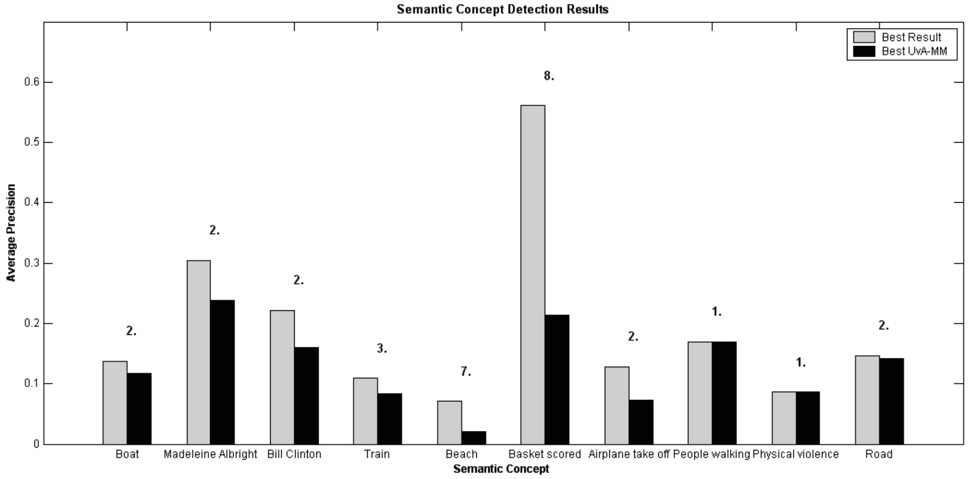


Figure 5: *Comparison of UvA-MM semantic concept detection results with other systems. In terms of system performance, UvA-MM ranks first in two concepts, second in five concepts, and third in one concept.*

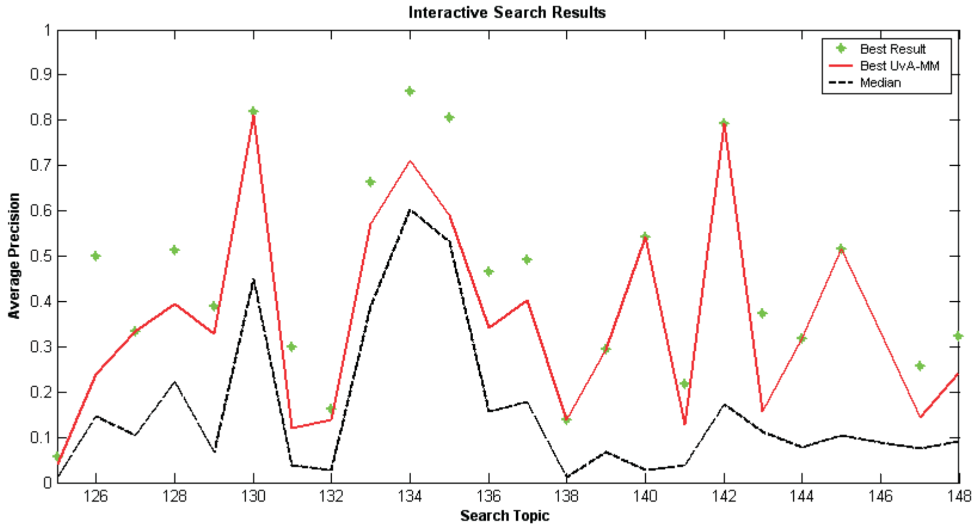


Figure 6: *Comparison of UvA-MM interactive search results with other systems. The UvA-MM runs rank first in seven topics. Furthermore the UvA-MM1 run has the best MAP (not visualized).*

search engine. For some topics there was a clear (accidental) overlap with the concepts in our lexicon, i.e. *ice hockey* in search topic 130, *bicycle* in search topic 140, and *Bill Clinton* in search topic 144. Not surprisingly we did very well on those topics. For others, general concept classes were available that allow to make a first selection, e.g. *sporting event* for tennis player in search topic 142 and *animal* for horses in search topic 145. Based on query by example and query by keyword, results can then be refined. Query by example is particularly useful when an answer to a search topic is found in a commercial. For search topics that did not have a clear overlap with the concepts in the lexicon, users had to rely on a combination of query by keyword and query by example. Here we performed less well, although still above the median. For example, wheelchairs in search topic 143 and Person X related topics that were not in the lexicon (128, 133, 134, 135, 137).

6 Conclusion

To bridge the semantic gap an ideal video retrieval system should combine query by concept and query by similarity in an interactive fashion. To that end we have developed a semantic video search engine. Main innovation is the possibility to query on a lexicon of 32 semantic concepts. The concepts are detected using the semantic value chain. The semantic value chain combines the content link, style link, and context link into a consecutive analysis chain, that allows for generic video indexing. Both the semantic value chain and the semantic video search engine are successfully evaluated within the 2004 TRECVID benchmark as top performers for their task.

The semantic value chain is as strong as its weakest link. For future research we plan to extend the semantic value chain with more advanced image and text analysis methods in the content link. Apart from improved content analysis we also plan to introduce feature selection in each link to optimize results. However, the greatest challenge ahead is to extend the lexicon of semantic concepts to a set that is compatible with human knowledge. This will have a dazzling impact on multimedia repository usage scenarios.

Acknowledgement

NIST for the evaluation effort. Kees Verstoep from the Vrije Universiteit Amsterdam for DAS-2 support. Jan van Gemert and Giang Nguyen from the University of Amsterdam for search experiment participation.

References

[1] A. Amir et al. IBM research TRECVID-2003 video retrieval system. In *Proc. of the TRECVID Workshop*, NIST Special Publication, Gaithersburg, USA, 2003.

[2] J. Baan et al. Lazy users and automatic video retrieval tools in (the) lowlands. In E. Voorhees and D. Harman, editors, *Proc. of the 10th Text REtrieval Conf.*, volume 500-250 of *NIST Special Publication*, Gaithersburg, USA, 2001.

[3] H. Bal et al. The distributed ASCI supercomputer project. *Operating Systems Review*, 34(4):76–96, 2000.

[4] 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/`.

[5] J. Fan, A. Elmagarmid, X. Zhu, W. Aref, and L. Wu. *ClassView*: hierarchical video shot classification, indexing, and accessing. *IEEE Trans. on Multimedia*, 6(1):70–86, 2004.

[6] J. Gauvain, L. Lamel, and G. Adda. The LIMSI broadcast news transcription system. *Speech Communication*, 37(1–2):89–108, 2002.

[7] J. Geusebroek, R. van den Boomgaard, A. Smeulders, and H. Geerts. Color invariance. *IEEE Trans. on PAMI*, 23(12):1338–1350, 2001.

[8] G. Iyengar, H. Nock, and C. Neti. Discriminative model fusion for semantic concept detection and annotation in video. In *ACM Multimedia*, pages 255–258, Berkeley, USA, 2003.

[9] C.-Y. Lin, B. Tseng, and J. Smith. Video collaborative annotation forum: Establishing ground-truth labels on large multimedia datasets. In *Proc. of the TRECVID Workshop*, NIST Special Publication, Gaithersburg, USA, 2003.

[10] M. Naphade. On supervision and statistical learning for semantic multimedia analysis. *Journal of Visual Communication and Image Representation*, 15(3):348–369, 2004.

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

[12] G. Salton and M. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill, New York, USA, 1983.

[13] T. Sato et al. Video OCR: Indexing digital news libraries by recognition of superimposed caption. *ACM Multimedia Systems*, 7(5):385–395, 1999.

[14] H. Schneiderman and T. Kanade. Object detection using the statistics of parts. *Int‘l Journal of Computer Vision*, 56(3):151–177, 2004.

[15] F. Seinstra, C. Snoek, D. Koelma, J. Geusebroek, and M. Worring. User transparent parallel processing of the 2004 NIST TRECVID data set. In *Int’l Parallel Distrib. Processing Symposium*, Denver, USA, 2005.

[16] A. Smeulders et al. Content based image retrieval at the end of the early years. *IEEE Trans. on PAMI*, 22(12):1349–1380, 2000.

[17] C. Snoek and M. Worring. Multimedia event based video indexing using time intervals. *IEEE Trans. on Multimedia*, 2005. In press.

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

[19] C. Snoek, M. Worring, and A. Hauptmann. Learning rich semantics from produced video. Submitted for publication.

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

[21] H. Wactlar, M. Christel, Y. Gong, and A. Hauptmann. Lessons learned from building a terabyte digital video library. *IEEE Computer*, 32(2):66–73, 1999.

[22] M. Worring et al. Interactive search using indexing, filtering, browsing and ranking. In *Proc. of the TRECVID Workshop*, NIST Special Publication, Gaithersburg, USA, 2003.

[23] Y. Wu, B. Tseng, and J. Smith. Ontology-based multi-classification learning for video concept detection. In *Proc. of the IEEE Int‘l Conf. on Multimedia & Expo*, Taipei, Taiwan, 2004.