The Semantic Pathfinder: Using an Authoring Metaphor for Generic Multimedia Indexing

Cees G.M. Snoek, *Student Member*, *IEEE*, Marcel Worring, *Member*, *IEEE*,
Jan-Mark Geusebroek, *Member*, *IEEE*, Dennis C. Koelma,
Frank J. Seinstra, *Member*, *IEEE*, and Arnold W.M. Smeulders, *Member*, *IEEE*

Abstract—This paper presents the semantic pathfinder architecture for generic indexing of multimedia archives. The semantic pathfinder extracts semantic concepts from video by exploring different paths through three consecutive analysis steps, which we derive from the observation that produced video is the result of an authoring-driven process. We exploit this *authoring metaphor* for machine-driven understanding. The pathfinder starts with the content analysis step. In this analysis step, we follow a data-driven approach of indexing semantics. The style analysis step is the second analysis step. Here, we tackle the indexing problem by viewing a video from the perspective of production. Finally, in the context analysis step, we view semantics in context. The virtue of the semantic pathfinder is its ability to learn the best path of analysis steps on a per-concept basis. To show the generality of this novel indexing approach, we develop detectors for a lexicon of 32 concepts and we evaluate the semantic pathfinder against the 2004 NIST TRECVID video retrieval benchmark, using a news archive of 64 hours. Top ranking performance in the semantic concept detection task indicates the merit of the semantic pathfinder for generic indexing of multimedia archives.

Index Terms—Video analysis, concept learning, benchmarking, content analysis and indexing, multimedia information systems, pattern recognition.

1 Introduction

QUERY-BY-KEYWORD is the paradigm on which machine-based text search is still based. Elaborating on the success of text-based search engines, query-by-keyword also gains momentum in multimedia retrieval. For multimedia archives, it is hard to achieve access, however, when based on text alone. Multimodal indexing is essential for effective access to video archives. For the automatic detection of specific concepts, the state-of-the-art has produced sophisticated and specialized indexing methods, see our previous work [1] and the work of Naphade and Huang [2] for an overview. Other than their textual counterparts, generic methods for semantic indexing in multimedia are neither generally available nor scalable in their computational needs nor robust in their performance. As a consequence, semantic access to multimedia archives is still limited. Therefore, there is a case to be made for a new approach to semantic video indexing.

The main problem for any semantic video indexing approach is the semantic gap between data representation and their interpretation by humans, as identified by Smeulders et al. [3]. In efforts to reduce the semantic gap, many video indexing approaches focus on specific semantic concepts with a small intraclass and large interclass variability of content. Typical concepts and their detectors are *sunsets* by Smith and Chang [4] and the work by Zhang et al. on *news anchors* [5]. These concepts have become icons for video indexing. Although they have aided in achieving progress, this approach is limited when considering the plethora of concepts waiting to be detected. It is simply impossible to bridge the semantic gap by designing a tailor-made solution for each concept.

In this paper, we propose a novel approach for generic semantic indexing of multimedia archives. It builds on the observation that produced video is the result of an authoring process. When producing a video, an author departs from a conceptual idea. The semantic intention is then articulated in (sub)consciously selected conventions and techniques for the purpose of emphasizing aspects of the content. The intention is communicated in context to the audience by a set of commonly shared notions. We aim to link the knowledge of years of media science research to semantic video analysis, see, for example, Boggs and Petrie [6] and Bordwell and Thompson [7]. We use the authoring-driven process of video production as the leading principle for generic video indexing.

Viewing semantic video indexing from an authoring perspective has the advantage that the most successful existing video indexing methods may be combined in one architecture. We first consider the vast amount of work performed in developing detection methods for specialized concepts [4], [5], [8], [9], [10], [11], [12]. If we measure the success of these methods in terms of benchmark detection performance, Informedia [8], [9] stands out. They focus on combining techniques from computer vision, speech recognition, natural language understanding, and artificial intelligence into a video indexing and retrieval environment. This has resulted in a large set of isolated and specialized concept detectors [9]. We build our generic

- *The authors are with the Intelligent Systems Lab Amsterdam, Informatics Institute, University of Amsterdam, Kruislaan 403, 1098 SJ Amsterdam, The Netherlands. E-mail: {cgmsnoek, worring, mark, koelma, fjseins, smeulders}@science.uva.nl.*

Manuscript received 1 Apr. 2005; revised 1 Jan. 2006; accepted 27 Feb. 2006; published online 11 Aug. 2006.
Recommended for acceptance by C. Schmid.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0173-0405.

indexing approach in part on the outputs of their detectors, but we do not use them in isolation.

In comparison with specialized detection methods, generic semantic indexing is rare. We discuss three successful examples of generic semantic indexing approaches [13], [14], [15]. In the first one, Fan et al. [13] propose the *ClassView* framework. The framework combines hierarchical semantic indexing with hierarchical retrieval. At the lowest level, the framework supports indexing of shots into concepts based on a large set of low-level visual features. At the second level, a Bayes classifier maps concepts to semantic clusters. By assigning shots to a hierarchy of concepts, the framework supports queries based on semantic and visual similarity. As the authors indicate, the framework would provide more meaningful results if it would support multimodal content analysis. We aim for generic semantic indexing also, but we include multimodal analysis from the beginning. In the second generic method [14], Amir et al. propose a system for semantic indexing using a detection pipeline. The pipeline starts with feature extraction, followed by consecutive aggregations on features, multiple modalities, and concepts. The pipeline optimizes the result by rule-based post filtering. We interpret the success of the system by the fact that all modules in the pipeline select the best of multiple hypotheses and the exhaustive use of machine learning. Moreover, the authors were among the first to recognize that semantic indexing profits substantially from context. We adopt and extend their ideas related to hypothesis selection, machine learning, and the use of context for semantic indexing. All of the above generic methods ignore the important influence of the video production style in the analysis process. In addition to content and context, we identify layout and capture in [15] as important factors for semantic indexing of produced video. We propose in [15] a generic framework for produced video indexing combining four sets of style detectors in an iterative semantic classifier. Results indicate that the method obtains high accuracy for rich semantic concepts, rich meaning that concepts share many similarities in their video production process. The framework is less suited for concepts that are not stylized. In the current paper, we generalize the idea of using style for semantic indexing.

We propose a generic approach for semantic indexing we call the *semantic pathfinder*. It combines the most successful methods for semantic video indexing [8], [9], [14], [15] into an integrated architecture. The design principle is derived from the video production process, covering notions of content, style, and context. The architecture is built on several detectors, multimodal analysis, hypothesis selection, and machine learning. The semantic pathfinder combines analysis steps at increasing levels of abstraction, corresponding to well-known facts from the study of film and television production [6], [7]. Its virtue is its ability to learn the best path, from all explored analysis steps, on a per-concept basis. To demonstrate the effectiveness of the semantic pathfinder, the semantic indexing experiments are evaluated within the 2004 NIST TRECVID video retrieval benchmark [16], [17].

The organization of this paper is as follows: First, we introduce the TRECVID benchmark in Section 2. Our system architecture for generic semantic indexing is presented in Section 3. We present results in Section 4.

2 TRECVID Benchmark

Evaluation of multimedia systems has always been a delicate issue. Due to copyrights and the sheer volume of data involved, multimedia archives are fragmented and mostly inaccessible. Therefore, comparison of systems has traditionally been difficult, often even impossible. To accommodate these hardships, NIST started organizing the TRECVID video retrieval benchmark. The benchmark aims to promote progress in video retrieval via open, metrics-based evaluation [16], [17]. Tasks include camera shot segmentation, story segmentation, semantic concept detection,[1] and several search tasks. Because of its widespread acceptance in the field, resulting in large participation of teams from both academic and corporate research labs worldwide, the benchmark can be regarded as the *de facto* standard to evaluate performance of multimedia indexing and retrieval research. We have participated in the semantic concept detection task of the 2004 NIST TRECVID video retrieval benchmark.

2.1 Multimedia Archive

The video archive of the 2004 TRECVID benchmark extends the data set used in 2003. The archive is composed of 184 hours of ABC World News Tonight and CNN Headline News and is recorded in MPEG-1 format. The training data consists of the archive used in 2003. It contains approximately 120 hours covering the period of January until June 1998. The 2004 test data contains the remaining 64 hours, covering the period of October until December 1998. Together with the video archive, CLIPS-IMAG [18] provided a camera shot segmentation. We evaluate semantic indexing within the TRECVID benchmark to demonstrate the effectiveness of the semantic pathfinder for semantic access to multimedia archives.

2.2 Evaluation Criteria

Participation in TRECVID is based on the submission of results for one or more of the concepts in the semantic concept detection task. Where a submission, or run, contains a ranked list of at most 2,000 camera shots per semantic concept and, for each concept, participants are allowed to submit up to 10 runs.

To determine the accuracy of submissions we use *average precision* and *precision at 100*, following the standard in TRECVID evaluations. The average precision is a single-valued measure that is proportional to the area under a recall-precision curve. This value is the average of the precision over all relevant judged shots. Hence, it combines precision and recall into one performance value. Let $L^k = \{l_1, l_2, \dots, l_k\}$ be a ranked version of the answer set A. At any given rank k, let $R \cap L^k$ be the number of relevant shots in the top k of L, where R is the total number of relevant shots. Then, average precision is defined as:

$$average\ precision = \frac{1}{R} \sum_{k=1}^{A} \frac{R \cap L^k}{k} \psi(l_k), \tag{1}$$

where indicator function $\psi(l_k) = 1$ if $l_k \in R$ and 0 otherwise. As the denominator k and the value of $\psi(l_k)$ are dominant in

[1] TRECVID refers to this task as the feature extraction task; to prevent misunderstanding with feature extraction as defined in the semantic pathfinder, we refer to it as the semantic concept detection task.

determining average precision, it can be understood that this metric favors highly ranked relevant shots.

TRECVID uses a pooled ground truth, P, to reduce labor-intensive manual judgments of all submitted runs. They take from each submitted run a fixed number of ranked shots, which is combined into a list of unique shots. Every submission is then evaluated based on the results of assessing this merged subset, i.e., instead of using R in (1), P is used, where $P \subset R$. This is a fair comparison for submitted runs since it assures that, for each submitted run, at least a fixed number of shots are evaluated at the more important top of the ranked list. However, using a pooled ground truth based on manual judgment comes with a price. In addition to mistakes by relevance assessors that may appear, using a pooling mechanism for evaluation means that the ground truth of the test data is incomplete.

Apart from average precision, we also report the precision at depth 100 in the result set. This value gives the fraction of correctly annotated shots within the first 100 retrieved results.

3 Semantic Pathfinder

Before we elaborate on the video indexing architecture, we first define a lexicon Λ_S of 32 semantic concepts. The lexicon is indicative for future efforts to detect as much as 1,000 concepts [19]. At present, it serves as a nontrivial illustration of concept possibilities. In addition, the anticipated positive influence of the lexicon on the result of the 10 benchmark concepts is taken into account. The semantic concept lexicon consists of the following concepts:

- $\Lambda_S =$ {*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*}.

The lexicon contains both general concepts, like *people*, *car*, and *beach*, as well as specific concepts, such as *airplane take off* and *news subject monologue*. We aim to detect all 32 concepts with the proposed system architecture.

The semantic pathfinder is composed of three analysis steps. It follows the reverse authoring process. Each analysis step in the path detects semantic concepts. In addition, one can exploit the output of an analysis step in the path as the input for the next one. The semantic pathfinder starts in the *content analysis step*. In this analysis step, we follow a data-driven approach of indexing semantics. The *style analysis step* is the second analysis step. Here, we tackle the indexing problem by viewing a video from the perspective of production. This analysis step especially aids in indexing of rich semantics. Finally, to enhance the indexes further, in the *context analysis step*, we view semantics in context. One would expect that some concepts, like *vegetation*, have their emphasis on content where the style (of the camera work that is) and context (of concepts like *graphics*) do not add much. In contrast, more complex events, like *people walking*, profit from incremental adaptation of the analysis to the intention of the author. The virtue of the semantic pathfinder is its ability to find the best path of analysis steps on a per-concept basis.

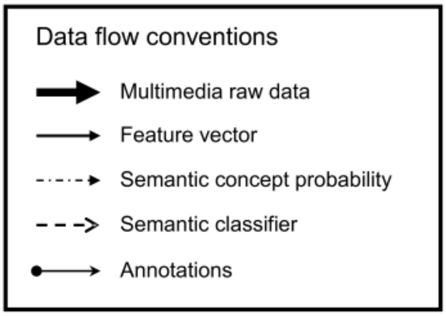


Fig. 1. Data flow conventions as used in this paper. Different arrows indicate difference in data flows.

The analysis steps in the semantic pathfinder exploit a common architecture, with a standardized input-output model, to allow for semantic integration. The conventions to describe the system architecture are indicated in Fig. 1. An overview of the semantic pathfinder is given in Fig. 2.

3.1 Analysis Step General Architecture

We perceive semantic indexing in video as a pattern recognition problem. We first need to segment a video. We opt for camera shots, indicated by i, following the standard in TRECVID evaluations. Given pattern x, part of a shot, the aim is to detect a semantic concept ω from shot i using probability $p(\omega|x_i)$. Each analysis step in the semantic pathfinder extracts x_i from the data and exploits a learning module to learn $p(\omega|x_i)$ for all ω in the semantic lexicon Λ_S. We exploit supervised learning to learn the relation between ω and x_i. The training data of the multimedia archive, together with labeled samples, are for learning classifiers. The other data, the test data, are set aside for testing. The general architecture for supervised learning in each analysis step is illustrated in Fig. 3.

Supervised learning requires labeled examples. In part, we rely on the ground truth provided in TRECVID 2003 [20]. We remove the many errors from this annotation effort. It is extended manually to arrive at an incomplete, but reliable ground truth[2] for all concepts in lexicon Λ_S. We split the training data a priori into a nonoverlapping training set and validation set to prevent overfitting of classifiers in the semantic pathfinder. It should be noted that a reliable validation set would ideally require an as large as possible a percentage of positively labeled examples, which is comparable to the training set. In practice, this may be hard to achieve, however, as some concepts are sparse. The training set we use contains 85 percent of the training data, the validation set contains the remaining 15 percent. We summarize the percentage of positively annotated examples for each concept in training and validation set in Table 1.

We choose from a large variety of supervised machine learning approaches to obtain $p(\omega|x_i)$. For our purpose, the method of choice should be capable of handling video documents. To that end, ideally, it must learn from a limited number of examples, it must handle unbalanced data, and it should account for unknown or erroneously detected data. In such heavy demands, the Support Vector Machine (SVM)

[2] http://www.science.uva.nl/~cgmsnoek/tv/.

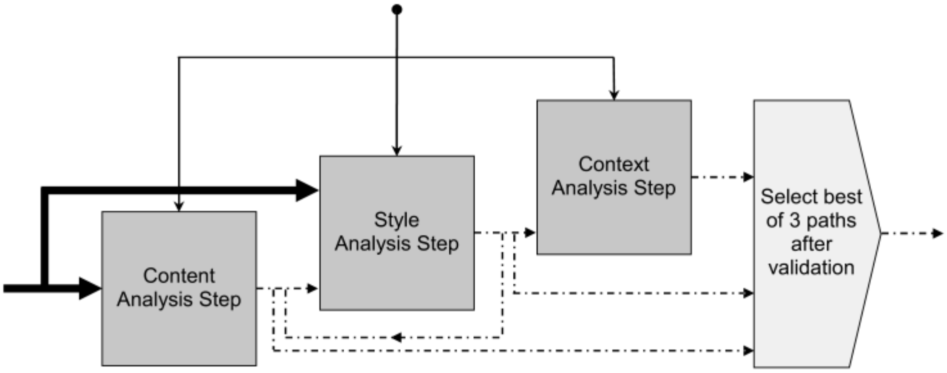


Fig. 2. The semantic pathfinder for one concept, using the conventions of Fig. 1.

framework [21], [22] has proven to be a solid choice [14], [23]. The usual SVM method provides a margin, $\gamma(x_i)$, in the result. We prefer Platt's conversion method [24] to achieve a posterior probability of the result. It is defined as:

$$p(\omega|x_i) = \frac{1}{1 + \exp(\alpha\gamma(x_i) + \beta)}, \tag{2}$$

where the parameters α and β are maximum likelihood estimates based on training data. SVM classifiers thus trained for ω, result in an estimate $p(\omega|x_i, \vec{q})$, where $\vec{q}$ are parameters of the SVM yet to be optimized.

The influence of the SVM parameters on concept detection is significant [25]. We obtain good parameter settings for a classifier by using an iterative search on a large number of SVM parameter combinations. We measure average precision performance of all parameter combinations and select the combination that yields the best performance, $\vec{q}^*$. Here, we use a three-fold cross validation [26] to prevent overfitting of parameters. The result of the parameter search over $\vec{q}$ is the improved model $p(\omega|x_i, \vec{q}^*)$, contracted to $p^*(\omega|x_i)$.

This concludes the introduction of the general architecture of all analysis steps in the semantic pathfinder.

3.2 Content Analysis Step

We view video in the content analysis step from the data perspective. In general, three data streams or modalities exist in video, namely, the auditory modality, the textual modality, and the visual one. As speech is often the most informative part of the auditory source, we focus on visual features and on textual features obtained from transcribed speech. After modality specific data processing, we combine features in a multimodal representation. The data flow in the content analysis step is illustrated in Fig. 4.

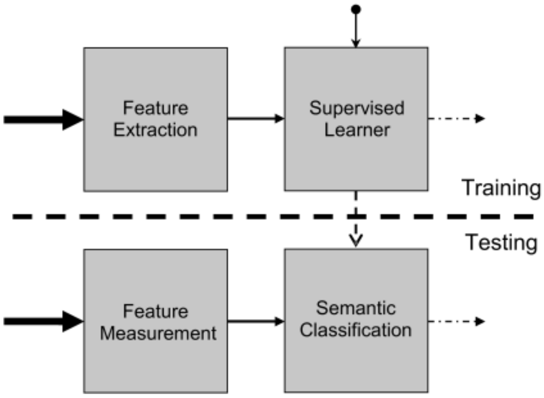


Fig. 3. General architecture of an analysis step in the semantic pathfinder, using the conventions of Fig. 1.

3.2.1 Visual Analysis

In the visual modality, we aim for segmentation of an image frame f into regional visual concepts. Ideally, a segmentation method should result in a precise partitioning of f according to the object boundaries, referred to as strong segmentation. However, weak segmentation, where f is partitioned into internally homogenous regions within the boundaries of the object, is often the best one can hope for [3]. We obtain a weak segmentation based on a set of visual feature detectors. Prior to segmentation we remove the border of each frame, including the space occupied by a possible ticker tape. The basis of feature extraction in the visual modality is weak segmentation.

Invariance was identified in [3] as a crucial aspect of a visual feature detector, e.g., to design features which limit the influence of accidental recording circumstances. We use color invariant visual features [27] to arrive at weak segmentation. The invariance covers the photometric variation due to shadow and shading and geometrical variation due to scale and orientation. This invariance is needed as the conditions under which semantic concepts appear in large multimedia archives may vary greatly.

The feature extraction procedure we adhere to computes, per pixel, a number of invariant features in vector $\vec{u}$. This vector then serves as the input for a multiclass SVM [22] that associates each pixel to one of the regional visual concepts defined in a visual concept lexicon Λ_V, using a labeled training set. Based on Λ_S, we define the following set of regional visual concepts:

- $\Lambda_V =$ {*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*}.

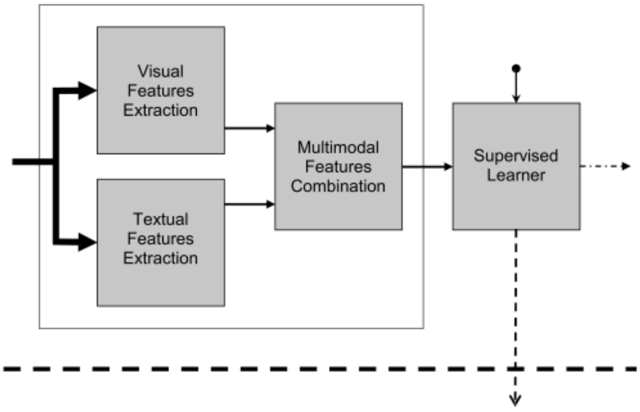
As we use invariant features, only a few examples per visual concept class are needed, in practice, less then 10 per class. This pixel-wise classification results in the image vector $\vec{w}_f$, where $\vec{w}_f$ contains one component per regional visual concept, indicating the percentage of pixels found for this class. Thus, $\vec{w}_f$ is a weak segmentation of frame f in terms of

TABLE 1
Semantic Concepts and the Percentage of Positively Labeled Examples Used for the Training Set and the Validation Set

Semantic Concept	*Training (%)*	*Validation (%)*	*Semantic Concept*	*Training (%)*	*Validation (%)*
Weather news	0.51	0.43	Golf	0.14	0.25
Stock quotes	0.26	0.30	People	3.89	3.99
News anchor	3.91	3.99	American football	0.05	0.10
Overlayed text	0.26	0.17	Outdoor	7.52	8.60
Basket scored	1.07	0.97	Car	1.57	2.10
Graphics	1.06	1.05	Bill Clinton	0.97	1.41
Baseball	0.74	0.66	News subject monologue	3.84	3.96
Sporting event	2.27	2.44	Animal	1.35	1.34
People walking	1.92	1.97	Road	1.44	1.98
Financial news anchor	0.35	0.35	Beach	0.42	0.61
Ice hockey	0.36	0.47	Train	0.21	0.36
Cartoon	0.60	0.73	Madeleine Albright	0.18	0.02
Studio setting	4.94	4.65	Building	4.95	4.81
Physical violence	2.73	3.14	Airplane take off	0.89	0.87
Vegetation	1.60	1.59	Bicycle	0.28	0.27
Boat	0.55	0.45	Soccer	0.06	0.09

regional visual concepts from Λ_V, see Fig. 5 for an example segmentation.

We use Gaussian color measurements to obtain $\vec{u}$ for weak segmentation [27]. We decorrelate RGB color values by linear transformation to the opponent color system [27]:



Visual Features Extraction
Textual Features Extraction
Multimodal Features Combination
Supervised Learner

Fig. 4. Feature extraction and classification in the content analysis step, special case of Fig. 3.

$$\begin{bmatrix} E \\ E_\lambda \\ E_{\lambda\lambda} \end{bmatrix} = \begin{pmatrix} 0.06 & 0.63 & 0.27 \\ 0.3 & 0.04 & -0.35 \\ 0.34 & -0.6 & 0.17 \end{pmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \tag{3}$$

Smoothing these values with a Gaussian filter, $G(\sigma)$, suppresses acquisition and compression noise. Moreover, we extract texture features by applying Gaussian derivative filters. We vary the size of the Gaussian filters, $\sigma = \{1, 2, 3.5\}$ to obtain a color representation that is compatible with variations in the target object size (leaving out pixel position parameters):

$$\begin{aligned} \hat{E}_j(\sigma) &= G_j(\sigma) * E, \\ \hat{E}_{\lambda j}(\sigma) &= G_j(\sigma) * E_\lambda, \\ \hat{E}_{\lambda\lambda j}(\sigma) &= G_j(\sigma) * E_{\lambda\lambda}, \end{aligned} \tag{4}$$

where $j \in \{\emptyset, x, y\}$ indicates either spatial smoothing or spatial differentiation and that, from now on, the hat symbol $(\hat{\cdot})$ implies a dependence on σ. Normalizing each opponent color value by its intensity suppresses global intensity variations. This results in two chromaticity values per color pixel:

$$\hat{C}_\lambda = \frac{\hat{E}_\lambda}{\hat{E}}, \quad \hat{C}_{\lambda\lambda} = \frac{\hat{E}_{\lambda\lambda}}{\hat{E}}. \tag{5}$$

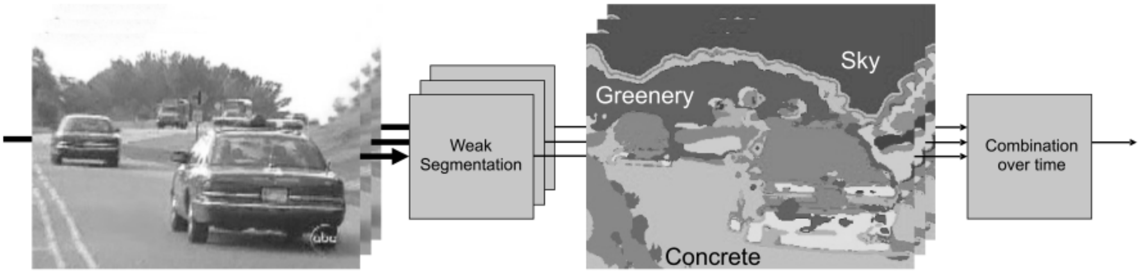


Fig. 5. Computation of the visual features, see Fig. 4, is based on weak segmentation of an image frame into regional visual concepts. A combination over time is used to select one frame as representative for the shot.

Furthermore, we obtain rotationally invariant features by taking Gaussian derivative filters and combining the responses into two chromatic gradients:

$$\hat{C}_{\lambda w} = \sqrt{\hat{C}^2_{\lambda x} + \hat{C}^2_{\lambda y}}, \quad \hat{C}_{\lambda\lambda w} = \sqrt{\hat{C}^2_{\lambda\lambda x} + \hat{C}^2_{\lambda\lambda y}}, \tag{6}$$

where $\hat{C}_{\lambda x}$, $\hat{C}_{\lambda y}$, $\hat{C}_{\lambda\lambda x}$, and $\hat{C}_{\lambda\lambda y}$ are defined as:

$$\begin{aligned} \hat{C}_{\lambda x} &= \frac{\hat{E}_{\lambda x}\hat{E} - \hat{E}_{\lambda}\hat{E}_{x}}{\hat{E}^2}, \; \hat{C}_{\lambda\lambda x} = \frac{\hat{E}_{\lambda\lambda x}\hat{E} - \hat{E}_{\lambda\lambda}\hat{E}_{x}}{\hat{E}^2}, \\ \hat{C}_{\lambda y} &= \frac{\hat{E}_{\lambda y}\hat{E} - \hat{E}_{\lambda}\hat{E}_{y}}{\hat{E}^2}, \; \hat{C}_{\lambda\lambda y} = \frac{\hat{E}_{\lambda\lambda y}\hat{E} - \hat{E}_{\lambda\lambda}\hat{E}_{y}}{\hat{E}^2}. \end{aligned} \tag{7}$$

The seven measurements computed in (4)-(6) and each calculated over three scales yield a 21-dimensional invariant feature vector $\vec{u}$ per pixel.

Segmenting image frames into regional visual concepts at the granularity of a pixel is computationally intensive. We estimate that the processing of the entire TRECVID data set would have taken around 250 days on the fastest sequential machine available to us. As a first reduction of the analysis load, we analyze 1 out of 15 frames only. For the remaining image processing effort, we apply the Parallel-Horus software architecture [28]. This architecture, consisting of a large collection of low-level image processing primitives, allows the programmer to write sequential applications with efficient parallel execution on commonly available commodity clusters. Application of Parallel-Horus, in combination with a distributed cluster consisting of 200 dual 1-Ghz Pentium-III CPUs [29], reduced the processing time to less than 60 hours [28].

The features over time are combined into one vector for the shot i. Averaging over individual frames is not a good choice as the visual representation should remain intact. Instead, we opt for a selection of the most representative frame or visual vector. To decide which f is the most representative for i, weak segmented image $\vec{w}_f$ is the input for an SVM that computes a probability $p^*(\omega|\vec{w}_f)$. We select $\vec{w}_f$ that maximizes the probability for a concept from Λ_S within i, given as:

$$\vec{v}_i = \arg\max_{f \in f_i} p^*(\omega|\vec{w}_f). \tag{8}$$

The visual vector $\vec{v}_i$, containing the best weak segmentation, is the final result of the visual analysis.

3.2.2 Textual Analysis

In the textual modality, we aim to learn the association between uttered speech and semantic concepts. A detection system transcribes the speech into text. From the text, we remove the frequently occurring stopwords. After stopword removal, we are ready to learn semantics.

To learn the relation between uttered speech and concepts, we connect words to shots. We make this connection within the temporal boundaries of a shot. We derive a lexicon of uttered words that co-occur with ω using the shot-based annotations of the training data. For each concept ω, we learn a separate lexicon, Λ^{ω}_{T}, as this uttered word lexicon is specific for that concept. We modify the procedure for Person X concepts, i.e., *Madeleine Albright* and *Bill Clinton*, to optimize results. In broadcast news, a news anchor or reporter mentions names or other indicative words just before or after a person is visible. To account for this observation, we stretch the shot boundaries with five seconds on each side for Person X concepts. For these concepts, this procedure assures that the textual feature analysis considers even more textual content. For feature extraction, we compare the text associated with each shot with Λ^{ω}_{T}. This comparison yields a text vector $\vec{t}_i$ for shot i, which contains the histogram of the words in association with ω.

3.2.3 Multimodal Analysis and Classification

The result of the content analysis step is a multimodal vector $\vec{m}_i$ that integrates all unimodal results. We concatenate the visual vector $\vec{v}_i$ with the text vector $\vec{t}_i$ to obtain $\vec{m}_i$. After this modality fusion, $\vec{m}_i$ serves as the input for the supervised learning module. To optimize parameter settings, we use three-fold cross validation on the training set. The content analysis step associates probability $p^*(\omega|\vec{m}_i)$ with a shot i, for all ω in Λ_S.

3.3 Style Analysis Step

In the style analysis step, we conceive of a video from the production perspective. Based on the four roles involved in the video production process [15], [30], this step analyzes a video by four related style detectors. Layout detectors analyze the role of the editor. Content detectors analyze the role of production design. Capture detectors analyze the role of the production recording unit. Finally, context detectors analyze the role of the preproduction team, see Fig. 6. Note that, in contrast to the content analysis step, where we learn specific content features from a data set, content features in the style analysis step are generic and independent of the data set.

3.3.1 Style Analysis

We develop detectors for all four production roles as feature extraction in the style analysis step. We refer to our

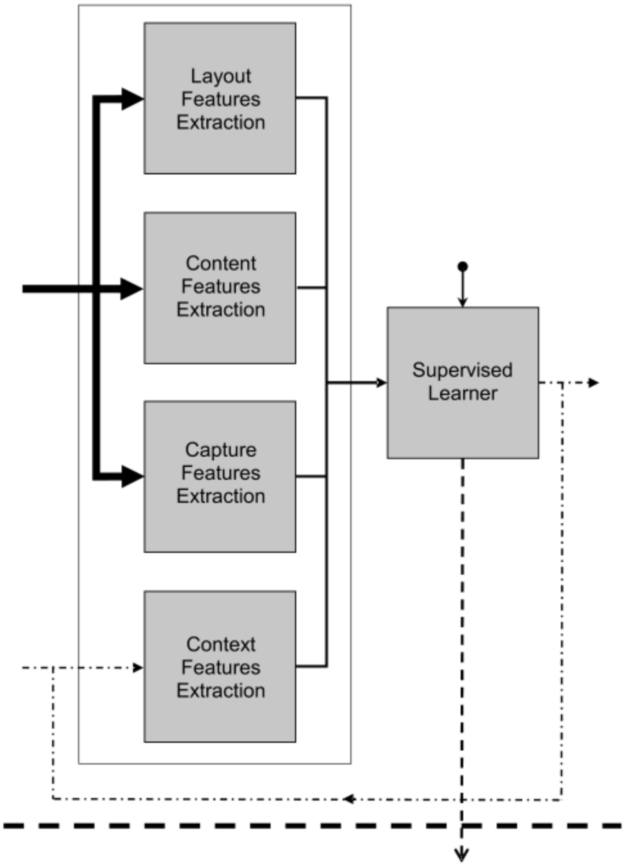

Layout Features Extraction
Content Features Extraction
Capture Features Extraction
Context Features Extraction
Supervised Learner

Fig. 6. Feature extraction and classification in the style analysis step, special case of Fig. 3.

previous work for specific implementation details of the detectors [15], [30, Appendix A]. We have chosen to convert the output of all style detectors to an ordinal scale 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 [1]. Overlayed text is another informative descriptor. Its presence is detected by a text localization algorithm [31]. To segment the auditory layout, periods of speech and silence are detected based on an automatic speech recognition system [32]. We obtain a voice-over detector by combining the speech segmentation with the camera shot segmentation [15]. The set of layout features is thus given by:

$$\mathcal{L} = \{shot\ length, overlayed\ text, silence, voice\text{-}over\}.$$

As concerns the content $\mathcal{C}$, a frontal face detector [33] is applied to detect people. We count the number of faces and, for each face, its location is derived [15]. Apart from faces, we also detect the presence of cars [33]. In addition, we measure the average amount of object motion in a camera shot [23]. Based on speaker identification [32], we 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 [15]. The length of text strings recognized by Video Optical Character Recognition [31] is used as a feature [15]. In addition, the strings are used as input for a named entity recognizer [8]. On the transcribed text obtained by the LIMSI automatic speech recognition system [32], 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 [33], [15]. It is undefined when no face is detected. In addition to camera distance, several types of camera work are detected [34], e.g., pan, tilt, zoom, etc. Finally, for capture, we also estimate the amount of camera motion [34]. The set of capture features is thus given by:

$$\mathcal{T} = \{camera\ distance, camera\ work, camera\ motion\}.$$

The context $\mathcal{S}$ serves to enhance or reduce the correlation between semantic concepts. Detection of *vegetation* can aid in the detection of a *forest* for example. Likewise, the co-occurrence of a *space shuttle* and a *bicycle* in one shot is improbable. As the performance of semantic concept detectors is unknown and likely to vary between concepts, we exploit iteration to add them to the context. The rationale here is to add concepts that are relatively easy to detect first. They aid in detection performance by increasing the number of true positives or reducing the number of false positives. As an initial concept, we detect news reporters. We recognize news reporters by edit distance matching of strings, obtained from the transcript and video text, with a database of names of CNN and ABC affiliates [15]. The other concepts that are added to the context stem from Λ_S. To prevent bias from domain knowledge, we use the performance on the validation set of all concepts from Λ_S in the content analysis step as the ordering for the context. For this ordering, we again refer to Table 1. To assign detection results for the first and least difficult concept, $\omega_1 = weather\ news$, we rank all shot results on $p_i^*(\omega_1|\vec{m}_i)$. This ranking is then exploited to categorize results for ω_1 into one of five levels. The basic set of context features is thus given by:

$$\mathcal{S} = \{news\ reporter, content\ analysis\ step\ \omega_1\}.$$

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

3.3.2 Iterative Style Classification

We start from an ordering of concepts in the context, as defined above. The iteration of the classifier begins with concept ω_1. After concatenation with the other style features, this yields $\vec{s}_{i,1}$, the first style vector of the first iteration. $\vec{s}_{i,1}$ contains the combined results of the content analysis step and the style analysis step. We classify ω_1 again based on $\vec{s}_{i,1}$. This yields the a posterior probability $p^*(\omega_1|\vec{s}_{i,1})$. When $p^*(\omega|\vec{s}_i) \geq \delta$, the concept ω_1 is considered present in the style representation, else it is considered absent. The threshold δ is set a priori at a fixed value of 0.5. In this process, the classifier replaces the feature for concept ω_1, from the content analysis step, by the new feature ω_1^+. The style analysis step adds more aspects of the author influence to the results obtained

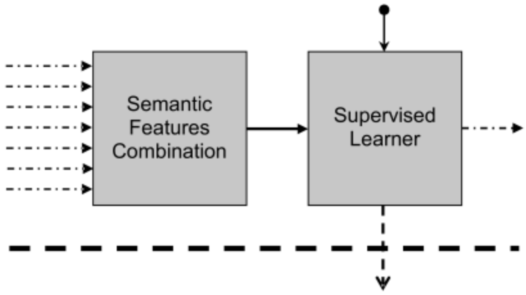


Fig. 7. Feature extraction and classification in the context analysis step, special case of Fig. 3.

with the content analysis step. In the next iteration of the classification procedure, the classifier adds $\omega_2 = stock\ quotes$ from the content analysis step to the context. This yields $\vec{s}_{i,2}$. As explained above, the classifier replaces the ω_2 feature from the content analysis step by the styled version ω_2^+ based on $p^*(\omega_2|\vec{s}_{i,2})$. This iterative process is repeated for all ω in lexicon Λ_S.

We classify all ω in Λ_S again in the style analysis step. As the result of the content analysis step is only one of the many features in our style vector representation in the style analysis step, we also use three-fold cross validation on the training set to optimize parameter settings in this analysis step. We use the resulting probability as output for concept detection in the style analysis step. In addition, it forms the input for the next analysis step in our semantic pathfinder.

3.4 Context Analysis Step

The context analysis step adds context to our interpretation of the video. Our ultimate aim is the reconstruction of the author's intent by considering detected concepts in context.

3.4.1 *Semantic Analysis*

The style analysis step yields a probability for each shot i and all concepts ω in Λ_S. The probability indicates whether a concept is present. We use the 32 concept scores as semantic features. We fuse them into context vector $\vec{c}_i$, see Fig. 7.

From $\vec{c}_i$, we learn relations between concepts automatically. To that end, $\vec{c}_i$ serves as the input for a supervised learning module, which associates a contextual probability $p^*(\omega|\vec{c}_i)$ to a shot i for all ω in Λ_S. To optimize parameter settings, we use three-fold cross validation on the previously unused data from the validation set.

The output of the context analysis step is also the output of the entire semantic pathfinder on video documents. On the way, we have included in the semantic pathfinder, the results of the analysis on raw data, facts derived from production by the use of style features, and a context perspective of the author's intent by using semantic features. For each concept, we obtain a probability based on content, style, and context. We select from the three possibilities the one that maximizes average precision based on validation set performance. The semantic pathfinder provides us with the opportunity to decide whether a one-shot analysis step is best for the concept only concentrating on content, or a two-analysis step classifier increasing discriminatory power by adding production style to content, or that a concept profits most from a consecutive analysis path using content, style, and context.

4 Results

4.1 Detection of 32 Semantic Concepts

We evaluated detection results for all 32 concepts in each analysis step. Given the already enormous size of the data sets and the large amounts of annotation—yet limited in terms of completeness—we have performed one pass for 32 concepts through the entire semantic pathfinder. We report the *precision at 100*, which indicates the number of correct shots within the first 100 results—assuming there are more than 100 relevant shots per concept—in Table 2.

We observe from the results that the learned best path (printed in bold) indeed varies over the concepts. The virtue of the semantic pathfinder is demonstrated by the fact that for 12 concepts, the learning phase indicates it is best to concentrate on content only. For five concepts, the semantic pathfinder demonstrates that a two-step path is best (where, in 15 cases, addition of style features has a marginal positive or negative effect). For 15 concepts, the context analysis step obtains a better result. Context aids substantially in the performance for 5 concepts. As an aside, we note that the precision at 100, when averaged over all concepts, steadily increases from 0.51 to 0.57 while traversing the different semantic analysis paths.

The results demonstrate the virtue of the semantic pathfinder. Concepts are divided by the analysis step after which they achieve best performance. Some concepts are just content, style does not affect them. In such cases as *American football*, there is, style-wise, too much confusion with other sports to add new value in the path. Shots containing *stock quotes* suffer from a similar problem. Here, false positives contain many stylistically similar results like graphical representations of survey and election results. For complex concepts, analysis based on content and style is not enough. They require the use of context. The context analysis step is especially good in detecting named events, like *people walking*, *physical violence*, and *basket scored*. The results offer us the possibility to categorize concepts according to the analysis step of the semantic pathfinder that yields the best performance.

The content analysis step seems to work particularly well for semantic concepts that have a small intraclass variability of content: *weather news* and *news anchor*, for example. In addition, this analysis step aids in detection of accidental content like *building*, *vegetation*, *bicycle*, and *train*. However, for some of those concepts, e.g., *bicycle* and *train*, the performance is still disappointing. Another observation is that, when one aims to distinguish subgenres, e.g., *ice hockey*, *baseball*, and *American football*, the content analysis step is the best choice.

After the style analysis step, we obtain an increase in performance for 12 concepts, see Fig. 8a. Especially when the concepts are semantically rich, e.g., *news subject monologue*, *financial news anchor*, and *sporting event*, the style helps. As expected, index results in the style analysis step improve on the content analysis step when style is a distinguishing property of the concept and degrade the result when similarity in style exists between different concepts.

Results after the context analysis step in Fig. 8b show that performance increases for 13 concepts. The largest positive performance difference between the context analysis step and the style analysis step occurs for concept *people*. Concept *people* profits from sport-related concepts like *baseball*, *basket scored*, *American football*, *ice hockey*, and *sporting event*. In contrast, *golf* suffers from detection of *outdoor* and *vegetation*.

TABLE 2
Test Set Precision at 100 after the Three Steps, for a Lexicon of 32 Concepts

Semantic Concept	*Content Analysis Step*	*Style Analysis Step*	*Context Analysis Step*	*Semantic Pathfinder*
News subject monologue	0.55	**1.00**	1.00	1.00
Weather news	**1.00**	1.00	1.00	1.00
News anchor	0.98	0.98	**0.99**	0.99
Overlayed text	0.84	**0.99**	0.93	0.99
Sporting event	0.77	**0.98**	0.93	0.98
Studio setting	0.95	0.96	**0.98**	0.98
Graphics	0.92	0.90	**0.91**	0.91
People	0.73	0.78	**0.91**	0.91
Outdoor	0.62	0.83	**0.90**	0.90
Stock quotes	**0.89**	0.77	0.77	0.89
People walking	0.65	0.72	**0.83**	0.83
Car	0.63	0.81	**0.75**	0.75
Cartoon	0.71	0.69	**0.75**	0.75
Vegetation	**0.72**	0.64	0.70	0.72
Ice hockey	**0.71**	0.68	0.60	0.71
Financial news anchor	0.40	**0.70**	0.71	0.70
Baseball	**0.54**	0.43	0.47	0.54
Building	**0.53**	0.46	0.43	0.53
Road	0.43	0.53	**0.51**	0.51
American football	**0.46**	0.18	0.17	0.46
Boat	0.42	0.38	**0.37**	0.37
Physical violence	0.17	0.25	**0.31**	0.31
Basket scored	0.24	0.21	**0.30**	0.30
Animal	0.37	0.26	**0.26**	0.26
Bill Clinton	**0.26**	0.35	0.37	0.26
Golf	**0.24**	0.19	0.06	0.24
Beach	0.13	0.12	**0.12**	0.12
Madeleine Albright	**0.12**	0.05	0.04	0.12
Airplane take off	0.10	0.08	**0.08**	0.08
Bicycle	0.09	**0.08**	0.07	0.08
Train	**0.07**	0.07	0.03	0.07
Soccer	**0.01**	0.01	0.00	0.01
Mean	*0.51*	*0.53*	*0.54*	*0.57*

The best result is given in bold. The corresponding path is selected in the semantic pathfinder.

When we detect *golf*, these concepts are also frequently present. The inverse, however, is not necessarily the case, i.e., when we detect *outdoor* it is not necessarily on a golf court. Based on these observations, we conclude that, apart from named events, detection results of the context analysis step are similar to those of the style analysis step. Index results improve based on presence of semantically related concepts, but the context analysis step is unable to capture the semantic structure between concepts and, for some concepts, this leads to a drop in performance.

The above results show that the semantic pathfinder facilitates generic video indexing. In addition, the semantic pathfinder provides the foundation of a technique taxonomy for solving semantic concept detection tasks. The fact that subgenres like *ice hockey*, *golf*, and *American football* behave similarly indicates the predictive value of the

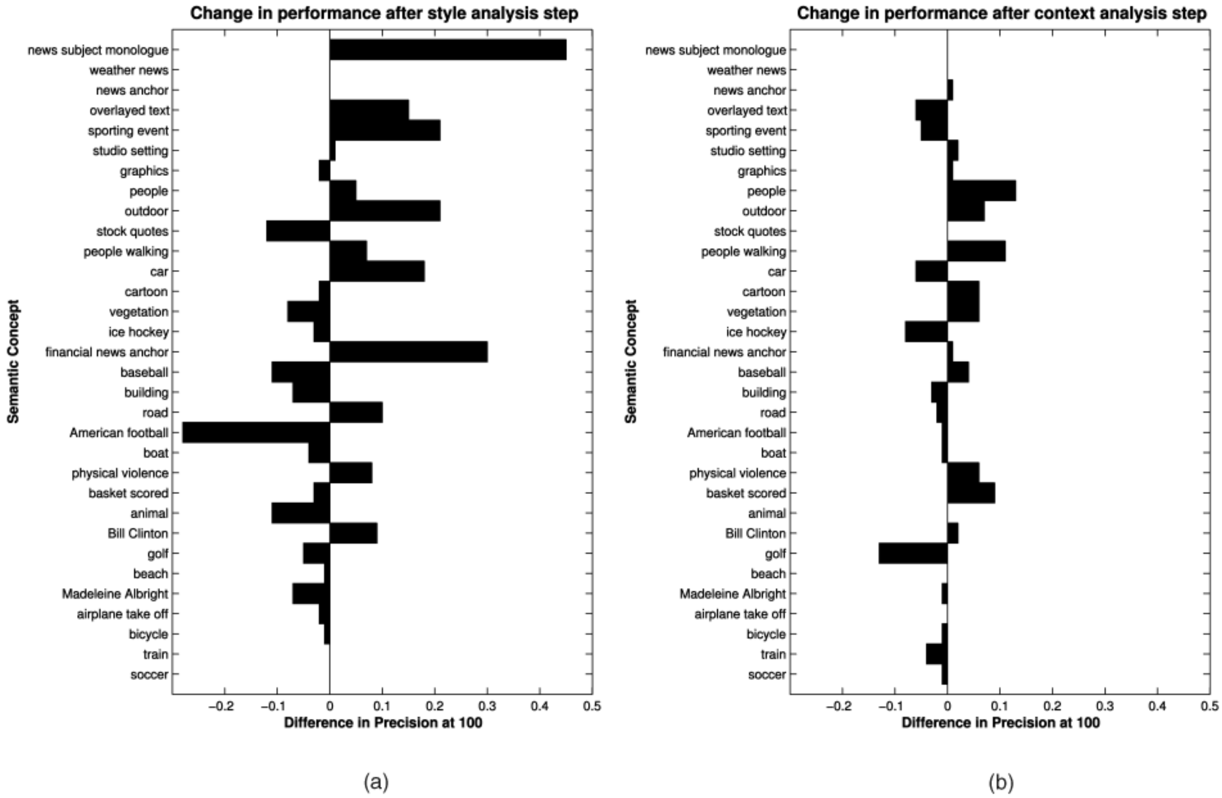


Fig. 8. Influence of (a) the style analysis step and (b) the context analysis step on precision at 100 performance for a lexicon of 32 semantic concepts. Note a considerable decrease (American football) or increase (news subject monologue) in performance when adding production style information. The same phenomenon is repeated for context information in golf (decrease) and people (increase).

pathfinder for other subgenres. The same holds for semantically rich concepts like *news subject monologue*, *financial news anchor*, and *sporting event*. We showed that, for named events, such as *basket scored*, *physical violence*, and *people walking*, one should apply a detector that is based on the entire semantic pathfinder. The significance of the semantic pathfinder is its generalizing power combined with the fact that the addition of new information in the analysis can be considered by concept type.

4.2 Benchmark Comparison

We performed an experiment within the TRECVID benchmark to show the effectiveness of the semantic pathfinder for detection of semantic concepts among 12 present-day video indexing systems. The TRECVID 2004 procedure prescribes that 10 predefined concepts are evaluated. Hence, we report the official benchmark results for 10 concepts in our lexicon only. The 10 benchmark concepts are, however, representative for the entire lexicon of 32. All evaluations are based on the semantic pathfinder.

We compare our work with the 11 other participants in TRECVID 2004. We select from each participant the system tuning with the best performance for a concept out of a maximum of 10 tunings. For ease of explanation we do not take the optimal tunings of the semantic pathfinder, as reported in [35], into account. Instead, we use a similar parameter setting for all concepts. Hence, we favor other systems in this comparison. Results are visualized in Fig. 9 for each concept.

Relative to other video indexing systems, the semantic pathfinder performs the best for two concepts, i.e., *people walking* and *physical violence*, and second for five concepts, i.e., *boat*, *Madeleine Albright*, *Bill Clinton*, *airplane take off*, and *road*. For two concepts we perform moderate, i.e., *basket scored* and *beach*. Here, the best approaches are based on specialized concept detection methods that exploit domain knowledge. The big disadvantage of these methods is that they are specifically designed and implemented for one concept. They do not scale to other concepts. The benchmark results show that the semantic pathfinder allows for generic indexing with state-of-the-art performance.

4.3 Usage Scenarios

The results from the semantic pathfinder facilitate the development of various applications. The lexicon of 32 semantic concepts allows for querying a video archive by concept. In [36], we combined into a semantic video search engine a query-by-concept, a query-by-keyword, a query-by-example, and an interactive filtering. In addition to interactive search, the set of indexes is also applicable in a personalized retrieval setting. A feasible scenario is that users with a specific interest in sports are provided with personalized summaries when and where they need it. The sketched applications provide a semantic access to multimedia archives.

5 Conclusion

We propose the semantic pathfinder for semantic access to multimedia archives. The semantic pathfinder is a generic

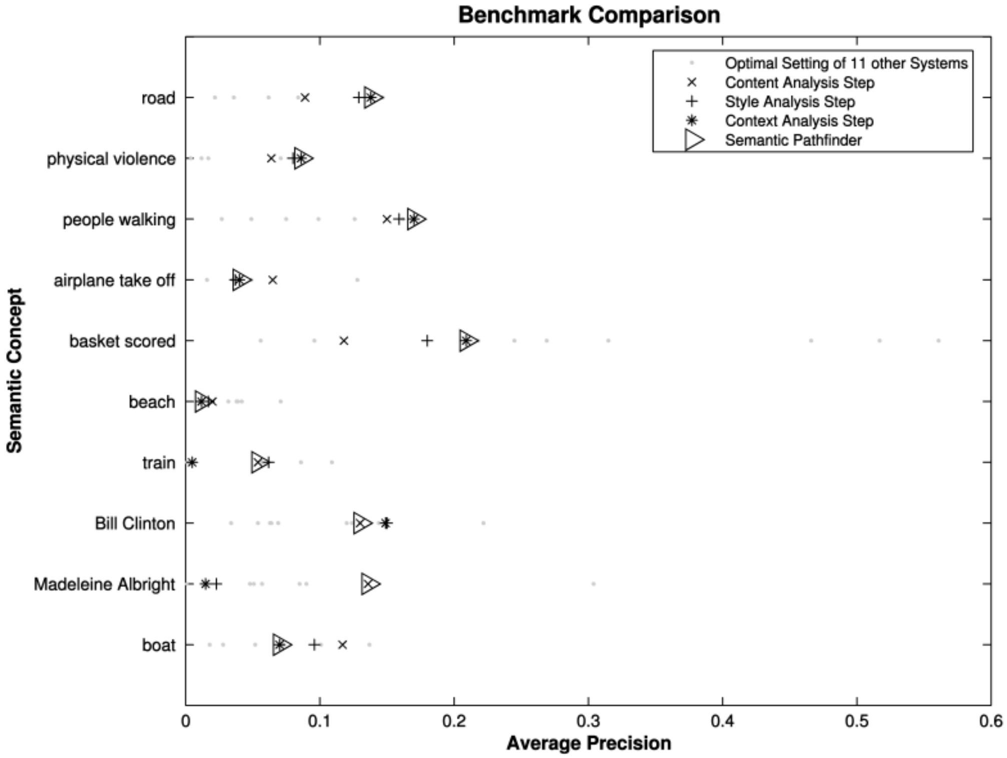


Fig. 9. Comparison of semantic pathfinder results with 11 other present-day indexing systems in the TRECVID 2004 benchmark [16], [17].

approach for video indexing. It is based on the observation that produced video is the result of an authoring process. The semantic pathfinder exploits the authoring metaphor in an effort to bridge the semantic gap. The architecture is built on a variety of detector types, multimodal analysis, hypothesis selection, and machine learning. The semantic pathfinder selects the best path through content analysis, style analysis, and context analysis. After machine learning, it appears that the analysis is completed after content analysis only when concepts share many similarities in their multimodal content. It also appears that the semantic path runs up to style analysis when the professional habits of television are evident to the concept. Finally, it exploits a path based on content, style, and context for concepts that are primarily intentional, see Table 2 and Fig. 8.

Experiments with a lexicon of 32 semantic concepts demonstrate that the semantic pathfinder allows for generic video indexing, while confirming the value of the authoring metaphor in indexing. In addition, the results over the various analysis steps indicate that a technique taxonomy exists for solving semantic concept detection tasks, depending on whether content, style, or context is most suited for indexing. Finally, the semantic pathfinder is successfully evaluated within the 2004 TRECVID benchmark. With one and the same set of system parameters two concepts, i.e., *people walking* and *physical violence*, came out best against 11 other present-day systems with average precision scores, remember that this measure indicates the average of the precision after every relevant item is retrieved, of 0.170 and 0.086, respectively. For five concepts, our system scored second best, i.e., *boat* (0.117), *Madeleine Albright* (0.136), *Bill Clinton* (0.150), *airplane take off* (0.065), and *road* (0.138). Just two performed poorly in this comparison, i.e., *basket scored* (0.209) and *beach* (0.020). The results show that the semantic pathfinder allows for state-of-the-art performance without the need for implementing specialized detectors. We consider this the best indication of the validity of the approach.

A semantic pathfinder is as strong as its weakest analysis step. Introduction of feature selection and knowledge representations in the various analysis steps will improve results. In its current form, the context analysis step takes the results of the style analysis step for granted; results are only adapted when there is enough contextual evidence from the other concepts to do so. Improvement of the semantic pathfinder along these lines is topic of future research.

For the moment, the average precision resulting from completely automatic indexing ranges from 0.020 to 0.209. In absolute terms, these performance values are still quite low. In 64 hours of produced video, only a small fraction of the relevant instances in the footage are retrieved within the first few ranked results. For selecting illustrative footage, this may already be sufficient. This is not yet so for tasks that require accurate retrieval. However, the trend in results over the past years indicates that automated search in video archives lures at the horizon.

ACKNOWLEDGMENTS

The authors are grateful to Howard Wactlar and the Informedia team from Carnegie Mellon University for their generous permission to include their software in the style analysis step. In addition, they acknowledge the benchmark organization and evaluation effort by NIST and the TRECVID coordinators. Finally, they thank Kees Verstoep from the Vrije Universiteit Amsterdam for DAS-2 support. This research is sponsored by TNO and the BSIK MultimediaN project.

REFERENCES

[1] C.G.M. Snoek and M. Worring, "Multimodal Video Indexing: A Review of the State-of-the-Art," *Multimedia Tools Applications,* vol. 25, no. 1, pp. 5-35, 2005.

[2] M.R. Naphade and T.S. Huang, "Extracting Semantics from Audiovisual Content: The Final Frontier in Multimedia Retrieval," *IEEE Trans. Neural Networks,* vol. 13, no. 4, pp. 793-810, 2002.

[3] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content Based Image Retrieval at the End of the Early Years," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 12, pp. 1349-1380, Dec. 2000.

[4] J.R. Smith and S.-F. Chang, "Visually Searching the Web for Content," *IEEE Multimedia,* vol. 4, no. 3, pp. 12-20, July-Sept. 1997.

[5] H.-J. Zhang, S.Y. Tan, S.W. Smoliar, and Y. Gong, "Automatic Parsing and Indexing of News Video," *Multimedia Systems,* vol. 2, no. 6, pp. 256-266, 1995.

[6] J.M. Boggs and D.W. Petrie, *The Art of Watching Films,* fifth ed. Mountain View, Calif.: Mayfield Publishing Company, 2000.

[7] D. Bordwell and K. Thompson, *Film Art: An Introduction,* fifth ed. McGraw-Hill, 1997.

[8] H.D. Wactlar, M.G. Christel, Y. Gong, and A.G. Hauptmann, "Lessons Learned from Building a Terabyte Digital Video Library," *Computer,* vol. 32, no. 2, pp. 66-73, Feb. 1999.

[9] A.G. Hauptmann et al., "Informedia at TRECVID 2003: Analyzing and Searching Broadcast News Video," *Proc. TRECVID Workshop,* 2003.

[10] N. Haering, R. Qian, and I. Sezan, "A Semantic Event-Detection Approach and Its Application to Detecting Hunts in Wildlife Video," *IEEE Trans. Circuits, Systems, and Video Technology,* vol. 10, no. 6, pp. 857-868, 2000.

[11] N. Babaguchi, Y. Kawai, and T. Kitahashi, "Event Based Indexing of Broadcasted Sports Video by Intermodal Collaboration," *IEEE Trans. Multimedia,* vol. 4, no. 1, pp. 68-75, 2002.

[12] A.A. Alatan, A.N. Akansu, and W. Wolf, "Multi-Modal Dialogue Scene Detection Using Hidden Markov Models for Content-Based Multimedia Indexing," *Multimedia Tools Applications,* vol. 14, no. 2, pp. 137-151, 2001.

[13] J. Fan, A.K. Elmagarmid, X. Zhu, W.G. Aref, and L. Wu, "ClassView: Hierarchical Video Shot Classification, Indexing, and Accessing," *IEEE Trans. Multimedia,* vol. 6, no. 1, pp. 70-86, 2004.

[14] A. Amir et al., "IBM Research TRECVID-2003 Video Retrieval System," *Proc. TRECVID Workshop,* 2003.

[15] C.G.M. Snoek, M. Worring, and A.G. Hauptmann, "Learning Rich Semantics from News Video Archives by Style Analysis," *ACM Trans. Multimedia Computing, Comm. Applications,* vol. 2, no. 2, pp. 91-108, May 2006.

[16] A.F. Smeaton, W. Kraaij, and P. Over, "The TREC VIDeo Retrieval Evaluation (TRECVID): A Case Study and Status Report," *Proc. Int'l Conf. Computer-Assisted Information Retrieval,* 2004.

[17] A.F. Smeaton, P. Over, and W. Kraaij, "TRECVID: Evaluating the Effectiveness of Information Retrieval Tasks on Digital Video," *ACM Multimedia,* 2004.

[18] G.M. Quénot, D. Moraru, L. Besacier, and P. Mulhem, "CLIPS at TREC-11: Experiments in Video Retrieval," *Proc. 11th Text REtrieval Conf.,* 2002.

[19] A.G. Hauptmann, "Towards a Large Scale Concept Ontology for Broadcast Video," *Proc. Third Int'l Conf. Image and Video Retrieval,* 2004.

[20] C.-Y. Lin, B.L. Tseng, and J.R. Smith, "Video Collaborative Annotation Forum: Establishing Ground-Truth Labels on Large Multimedia Datasets," *Proc. TRECVID Workshop,* 2003.

[21] V.N. Vapnik, *The Nature of Statistical Learning Theory,* second ed. Springer-Verlag, 2000.

[22] C.-C. Chang and C.-J. Lin, *LIBSVM: A Library For Support Vector Machines,* 2001, http://www.csie.ntu.edu.tw/cjlin/libsvm/.

[23] C.G.M. Snoek and M. Worring, "Multimedia Event-Based Video Indexing Using Time Intervals," *IEEE Trans. Multimedia,* vol. 7, no. 4, pp. 638-647, 2005.

[24] J.C. Platt, "Probabilities for SV Machines," *Advances in Large Margin Classifiers,* pp. 61-74, 2000.

[25] M.R. Naphade, "On Supervision and Statistical Learning for Semantic Multimedia Analysis," *J. Visual Comm. Image Representation,* vol. 15, no. 3, pp. 348-369, 2004.

[26] A.K. Jain, R.P.W. Duin, and J. Mao, "Statistical Pattern Recognition: A Review," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 1, pp. 4-37, Jan. 2000.

[27] J.M. Geusebroek, R. van den Boomgaard, A.W.M. Smeulders, and H. Geerts, "Color Invariance," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 12, pp. 1338-1350, Dec. 2001.

[28] F.J. Seinstra, C.G.M. Snoek, D. Koelma, J.M. Geusebroek, and M. Worring, "User Transparent Parallel Processing of the 2004 NIST TRECVID Data Set," *Proc. Int'l Parallel Distribution Processing Symp.,* 2005.

[29] H.E. Bal et al., "The Distributed ASCI Supercomputer Project," *Operating System Rev.,* vol. 34, no. 4, pp. 76-96, 2000.

[30] C.G.M. Snoek, "The Authoring Metaphor to Machine Understanding of Multimedia," PhD dissertation, Univ. of Amsterdam, 2005, http://www.science.uva.nl/~cgmsnoek/pub/snoek-thesis.pdf.

[31] T. Sato, T. Kanade, E.K. Hughes, M.A. Smith, and S. Satoh, "Video OCR: Indexing Digital News Libraries by Recognition of Superimposed Caption," *Multimedia Systems,* vol. 7, no. 5, pp. 385-395, 1999.

[32] J.L. Gauvain, L. Lamel, and G. Adda, "The LIMSI Broadcast News Transcription System," *Speech Comm.,* vol. 37, nos. 1-2, pp. 89-108, 2002.

[33] H. Schneiderman and T. Kanade, "Object Detection Using the Statistics of Parts," *Int'l J. Computer Vision,* vol. 56, no. 3, pp. 151-177, 2004.

[34] J. Baan et al., "Lazy Users and Automatic Video Retrieval Tools in (the) Lowlands," *Proc. 10th Text REtrieval Conf.,* E.M. Voorhees and D.K. Harman, eds., pp. 159-168, 2001.

[35] C.G.M. Snoek, M. Worring, J.M. Geusebroek, D.C. Koelma, and F.J. Seinstra, "The MediaMill TRECVID 2004 Semantic Video Search Engine," *Proc. TRECVID Workshop,* 2004.

[36] C.G.M. Snoek et al., "MediaMill: Exploring News Video Archives Based on Learned Semantics," *Proc. ACM Multimedia Conf.,* pp. 225-226, 2005.

Cees G.M. Snoek received the MSc degree in business information systems (2000) and the PhD degree in computer science (2005) both from the University of Amsterdam, where he is currently a senior researcher at the Intelligent Systems Lab. He was a visiting scientist at Informedia, Carnegie Mellon University, in 2003. His research interests focus on multimedia signal processing, statistical pattern recognition, content-based information retrieval, and large-scale benchmark evaluations, especially when applied in combination for multimedia understanding. Dr. Snoek is a lead architect of the MediaMill video search engine, which obtained state-of-the-art performance in recent NIST TRECVID evaluations and was awarded best technical demonstration at ACM Multimedia 2005. He is the local chair of the 2007 International Conference on Image and Video Retrieval in Amsterdam. He is a student member of the IEEE.

Marcel Worring received the MSc degree (honors) and PhD degree, both in computer science, from the Vrije Universiteit, Amsterdam, The Netherlands, in 1988 and the University of Amsterdam in 1993, respectively. He is currently an associate professor at the University of Amsterdam. His interests are in multimedia search and systems. He leads several multidisciplinary projects covering knowledge engineering, pattern recognition, image and video analysis, and information space interaction, conducted in close cooperation with industry. In 1998, he was a visiting research fellow at the University of California, San Diego. He has published more than 50 scientific papers and serves on the program committee of several international conferences. He is the chair of the IAPR TC12 on Multimedia and Visual Information Systems. He is general chair of the 2007 International Conference on Image and Video Retrieval in Amsterdam. He is a member of the IEEE.

Jan-Mark Geusebroek received the PhD degree in computer science from the University of Amsterdam in 2000. He is an assistant professor at the Intelligent Systems Lab Amsterdam, University of Amsterdam. He was awarded the E.S. Gelsema prize for his outstanding thesis. Furthermore, he received a prestigious young talent grant—a VENI Innovational Research Incentive—from the Netherlands Organization for Scientific Research. His current research interest is in cognitive computer vision, especially front-end color and texture vision, and mechanisms of focal attention. He is a member of the IEEE.

Dennis C. Koelma received the MSc and PhD degrees in computer science from the University of Amsterdam in 1989 and 1996, respectively. The subject of his thesis is a software environment for image interpretation. Currently, he is working on Horus, a software architecture for doing research in accessing the content of digital images and video. His research interests include image and video processing, software architectures, parallel programming, databases, graphical user interfaces, and image information systems.

Frank J. Seinstra received the MSc degree in computer science from the Vrije Universiteit in Amsterdam in 1996 and the PhD degree in computer science from the University of Amsterdam in 2003. Currently, he is a senior researcher at the Intelligent Systems Lab, University of Amsterdam. In addition, he is a visiting lecturer at the Academic Medical Center, Amsterdam, as well as a visiting researcher at the Computer Systems Group, Vrije Universiteit, Amsterdam. His research interests include parallel, distributed, and Grid computing, automatic parallelization, performance modeling, and scheduling, with a main focus on the application area of multimedia computing. He is a member of the IEEE.

Arnold W.M. Smeulders received the MSc degree in physics from the Technical University of Delft in 1977 and the PhD degree in medicine from Leiden University in 1982 on the topic of visual pattern analysis. He is scientific director of the Intelligent Systems Lab Amsterdam, of the MultimediaN the Dutch public-private partnership, and of the ASCI national research school. He participates in the EU-Vision, DELOS, and MUSCLE networks of excellence. He is a fellow of the International Association of Pattern Recognition. His research interest is in cognitive vision, content-based image retrieval, learning and tracking, and the picture-language question. He has written 300 papers in refereed journals and conferences and graduated 28 PhD students. The ISIS research group concentrates on theory, practice, and implementation of multimedia information analysis including image databases and computer vision. The group has an extensive record in cooperation with Dutch institutions and industry in the area of multimedia and video analysis. Currently, he is an associate editor of the *International Journal of Computer Vision* and the *IEEE Transactions on Multimedia*. He is a member of the IEEE.