Video Annotation and Retrieval Using Ontologies and Rule Learning

Lamberto Ballan, Marco Bertini, Alberto Del Bimbo, and Giuseppe Serra
University of Florence, Italy

An approach for automatic annotation and retrieval of video content uses semantic concept classifiers and ontologies to permit expanded queries to synonyms and concept specializations.

While understanding the semantic meaning of video content is immediate for humans, it's far from immediate for a computer. This discrepancy is commonly referred to as the *semantic gap*. A recent trend in the effort to bridge this gap is to define a large set of semantic concept detectors, each of which automatically detects the presence of a semantic concept such as ''indoor,'' ''face,'' ''person,'' or ''airplane flying.'' Typically these detectors learn the mapping between a set of low-level visual features, such as local descriptors, color and texture, and a concept from examples. Approaches to bridge this gap (see the ''Related Work'' sidebar) have been implemented in several systems designed for visual-concept recognition challenges, such as the Pascal Video Object Classes Challenge[1] and the Text Retrieval Conference Video (Trecvid) retrieval evaluation.[2] Generally, however, semantic concepts are still difficult to detect accurately, so their detection in video remains a challenging problem.

The accuracy of state-of-the-art detection can range from less than 0.1 (measured by average precision) for semantic concepts such as ''people marching'' or ''fire weapon'' to above 0.6 for a concept such as ''face.'' Despite the fact that performance improvements have been reported in the last years, and a large effort has been devoted to extend the number of different concept classifiers, important questions remain, such as how many concept detectors are really useful[3] and how reliable they should be.[4] Moreover, concept classifiers are usually drawn from a particular domain, but an important question is how well they can generalize across different domains.

Exploiting the semantic relationships between concepts is receiving a large amount of attention from the scientific community because doing so can improve the detection accuracy of concepts and obtain a richer semantic annotation of a video. To this end, ontologies are expected to improve the capability of computer systems to automatically detect even complex concepts and events from visual data with higher reliability. Ontologies consist of concepts, concept properties, and relationships between concepts. They organize semantic heterogeneity of information, using a formal representation, and provide a common vocabulary that encodes semantics and supports reasoning.

There have been few attempts to integrate high-level semantic concepts provided by an ontology with their visual representation. In the most common approach, the ontology provides the conceptual view of the domain at the schema level, and appropriate concept detectors play the role of observers of the real-world sources, classifying an observed entity or event in the nearest concept of the ontology. In this way, concept detectors have the responsibility of implementing invariance with respect to several conditions while, once the observations are classified, the ontology is exploited to have a more complete semantic annotation, establishing links to other concepts and disambiguating the results of classification.

In real applications, there is need to detect and recognize complex concepts and situations where multiple elementary concepts are in mutual relation in time and space. Therefore, ontologies have to be extended to define these higher-level concepts, adding sets of rules that encode spatiotemporal relationships among individual concepts. As the number of these concepts grows, the number of rules for their detection increases. Thus, the definition of rules by human experts is not practical; the

1070-986X/10/$26.00 © 2010 IEEE

Published by the IEEE Computer Society

Related Work

The usefulness of the construction of large sets of automatic video concept classifiers and the evaluation of the number of detectors needed for effective video retrieval has been studied in several works.[1-3] Hauptmann et al. report that concept-based video retrieval (with fewer than 5,000 concepts detected) with a minimal 0.1 mean average precision is likely to provide high accuracy in news video retrieval.[1] Snoek and Worring confirmed the positive correlation between the number of concept detectors and video retrieval performance, as well as the improvement of the pairwise combination of detectors, using a set of 363 concept detectors.[3]

Presently, the performance of video search engines is still far from acceptable.[4-6] In fact, their performance in terms of mean average precision, obtained in the Trecvid 2008 evaluation using 20 concepts from the Large Scale Concept Ontology for Multimedia (LSCOM) lexicon, varies in a range from 0.19 to 0.13. Ontologies and concept relations have been recently proposed to improve the performance of the concept detectors. Zha et al.[7] defined an ontology to provide a simple structure to LSCOM,[8] using pairwise correlations between concepts and hierarchical relationships to refine concept detection of support vector machine classifiers. Wei et al. have proposed two semantic spaces, ontology-enriched semantic space and ontology-enriched orthogonal semantic space, to facilitate the selection and fusion of concept detectors for video search.[9] In a different approach, proposed by Bertini et al., the ontology includes visual data instances related to high-level concepts, identifying their spatiotemporal patterns; visual prototypes, representative of these patterns, are then defined and used for automatic annotation.[10]

To obtain richer annotations, other authors have explored the use of rule-based reasoning over objects and events in different domains. Hollink et al. defined a set of SWRL rules to perform semiautomatic annotation of images of pancreatic cells.[11] Bai et al. defined a soccer ontology and applied temporal reasoning, with temporal description logic, to perform event annotation in soccer videos.[12] All these approaches expect that rules are created by human experts; thus, they are not practical for the definition of a large set of rules.

Automatic learning of rules has been proposed by Shyu et al.[13] These authors proposed a method to annotate rare events and concepts based on a set of rules that use low-level and middle-level features. A decision-tree algorithm is applied to the rule-learning process. Moreover, they addressed the imbalance problem of positive and negative examples in the case of rare events and concepts using data-mining techniques. Liu et al. proposed a method to enhance the accuracy of semantic concept detection using association-mining techniques to imply the presence of a concept from the co-occurrence of other high-level concepts.[14] However, these methods have shown to be insufficiently expressive to describe composite concepts and events because they don't take into account spatiotemporal relations between individual concepts.

References

1. A. Hauptmann et al., ''Can High-Level Concepts Fill the Semantic Gap in Video Retrieval? A Case Study with Broadcast News,'' *IEEE Trans. Multimedia,* vol. 9, no. 5, 2007, pp. 958-966.
2. J. Yang and A. Hauptmann, ''(Un)reliability of Video Concept Detection,'' *Proc. ACM Int'l Conf. Image and Video Retrieval,* ACM Press, 2008, pp. 85-94.
3. C. Snoek and M. Worring, ''Are Concept Detector Lexicons Effective for Video Search?'' *Proc. IEEE Int'l Conf. Multimedia & Expo,* IEEE Press, 2007, pp. 1966-1969.
4. C. Snoek et al., ''The MediaMill Trecvid 2008 Semantic Video Search Engine,'' *Proc. 6th Trecvid Workshop,* 2008; http://www-nlpir.nist.gov/projects/tvpubs/tv7.papers/mediamill.pdf.
5. S.F. Chang et al., ''Columbia University/Vireo-CityU/IRIT Trecvid2008 High-Level Feature Extraction and Interactive Video Search,'' *Proc. 6th Trecvid Workshop,* 2008; http://www-nlpir.nist.gov/projects/tvpubs/tv8.papers/columbia.pdf.
6. A. Natsev et al., ''IBM Research Trecvid-2008 Video Retrieval System,'' *Proc. 6th Trecvid Workshop,* 2008; http://www-nlpir.nist.gov/projects/tvpubs/tv8.papers/ibm.pdf.
7. Z.-J. Zha et al., ''Building a Comprehensive Ontology to Refine Video Concept Detection,'' *Proc. ACM Int'l Workshop Multimedia Information Retrieval,* ACM Press, 2007, pp. 227-236.
8. M. Naphade et al., ''Large-Scale Concept Ontology for Multimedia,'' *IEEE MultiMedia,* vol. 13, no. 3, 2006, pp. 86-91.
9. X.-Y. Wei, C.-W. Ngo, and Y.-G. Jiang, ''Selection of Concept Detectors Using Ontology-Enriched Semantic Space,'' *IEEE Trans. Multimedia,* vol. 10, no. 6, 2008, pp. 1085-1096.
10. M. Bertini et al., ''Dynamic Pictorially Enriched Ontologies for Digital Video Libraries,'' *IEEE MultiMedia,* vol. 16, no. 2, 2009, pp. 42-51.
11. L. Hollink, S. Little, and J. Hunter, ''Evaluating the Application of Semantic Inferencing Rules to Image Annotation,'' *Proc. Int'l Conf. Knowledge Capture,* ACM Press, 2005, pp. 91-98.
12. L. Bai et al., ''Video Semantic Content Analysis Based on Ontology,'' *Proc. Int'l Machine Vision and Image Processing Conf.,* IEEE Press, 2007, pp. 117-124.
13. M.-L. Shyu et al., ''Video Semantic Event/Concept Detection Using a Subspace-Based Multimedia Data Mining Framework,'' *IEEE Trans. Multimedia,* vol. 10, no. 2, 2008, pp. 252-259.
14. K.-H. Liu et al., ''Association and Temporal Rule Mining for Post-Filtering of Semantic Concept Detection in Video,'' *IEEE Trans. Multimedia,* vol. 10, no. 2, 2008, pp. 240-251.

appropriate solution is to learn a set of rules automatically for each composite concept to be detected.

In this article, we present an approach for automatic annotation and retrieval of video content based on ontologies and semantic-concept classifiers. First of all, automatic determination of semantic linguistic relations between concepts (is a, has part, is part of) is performed, using WordNet, to define the ontology schema; the concept detectors are then linked to the corresponding concepts in the ontology. We propose a novel, rule-based method for automatic semantic annotation of composite concepts and events in videos. Our algorithm learns rules expressed in Semantic Web Rules Language (SWRL) automatically, exploiting the knowledge embedded in the ontology. Moreover, the concepts' relationship of co-occurrence and the temporal consistency of video data are used to improve the performance of individual concept detectors. Finally, we present a Web video search engine that relies on ontologies and permits queries using a composition of Boolean and temporal relations between concepts. This system exploits the ontology structure and permits, for example, expanded queries to synonyms and concept specializations.

Automatic rule learning with first-order logic

In our approach, first-order logic rules defined in SWRL are automatically learned from the knowledge that is embedded in the ontology. Our ontology contains abstract concepts, the ontology schema (based on concepts detected by semantic classifiers and their linguistic relations as encoded in WordNet), and, for each concept, a set of the concept instances that have been observed. Rules are learned using first-order inductive learner for SWRL (Foils), a new algorithm obtained as an adaptation of the first-order inductive learner (FOIL[5]) technique to ontologies and Semantic Web technologies.

All the expressions are composed of constants, variables, predicate symbols, and function symbols. The difference between predicates and functions is that predicates (in the following written with an upper-case first letter) can assume only Boolean values, whereas functions (in the following written in lowercase) might have any constant as their value. A *term* is any constant, any variable, or any function. A *literal* is any predicate, or its negation, applied to any term. If a literal contains a negation symbol ($\neg$), it's called *negative literal*, otherwise it's a *positive literal*. A *clause* is any disjunction of literals, where all variables are assumed to be universally quantified.

A *Horn clause* is a clause containing at most one positive literal, as in: $H \vee \neg L_1 \vee \neg L_2 \ldots \vee \neg L_n$, where H is the positive literal, and $\neg L_1 \vee \neg L_2 \ldots \vee \neg L_n$ are negative literals. It is equivalent to: $(L_1 \wedge L_2 \ldots \wedge L_n) \rightarrow H$, which is equivalent to "IF $(L_1 \vee L_2 \ldots L_n)$ THEN H". The Horn clause precondition $(L_1 \wedge L_2 \ldots L_n)$ is called *body*, while the literal H that forms the postcondition, is called *head*. As an example of the Horn clause, consider the sentence that describes the composite concept: "a person is in a secured area" (IF a person and a secured area instances occur in a shot and the bounding box of that person is in the bounding box of that secured area THEN that person is in secured area). This sentence can be translated in the following fragment in first-order logic:

$$\begin{aligned} &Person\ (p) \wedge SecuredArea\ (s) \\ &\quad \wedge HasBoundingBox\ (p, pBox) \\ &\quad \wedge HasBoundingBox\ (s, sBox) \\ &\quad \wedge BoxIsInBox\ (pBox, sBox) \\ &\quad \rightarrow PersonIsInSecuredArea\ (p) \end{aligned}$$

where p and s are variables that can be bound to any person and any secured area respectively, while *sBox* and *pBox* are their bounding boxes.

The hypotheses learned by Foils are sets of rules that are Horn clauses. The algorithm starts with an initial rule, written in SWRL, composed of the head (that is, the target composite concept) and an empty or initial body, and an ontology with a set of instances that are positive and negative examples of the target concept. As an example, the initial rule for the composite concept "a person enters in a secured area" could be: $Person(p) \wedge SecuredArea(s) \rightarrow PersonEntersSecuredArea(p)$. The algorithm iterates searching new literals that have to be added to the body. This is a general-to-specific search through the space of hypotheses, beginning with the most general preconditions possible (the empty or initial precondition), and adding literals one at a time to specialize the rule until it avoids all negative examples, or when no more negative examples are excluded for a certain number of iterations l. A schema of the algorithm is shown Figure 1.

Two issues have to be addressed: the generation of hypothesis candidates and the choice of the most promising candidate. Suppose that at the ith iteration, the current rule R_i being considered is $(L_1 \wedge L_2 \ldots \wedge L_n) \rightarrow H(x_1, x_2, \ldots, x_k)$, where $(L_1 \wedge L_2 \ldots \wedge L_n)$ are literals forming the current rule preconditions and $H(x_1, x_2, \ldots, x_k)$ is the head. Foils generates candidate specializations of this rule by considering as new literals L_{i+1} any predicate occurring in the ontology (that is, all concepts and concepts relations), where at least a variable already exists in the rule. A special literal, $Equal(x_j, x_k)$ where x_j and x_k are variables already present in the rule, can be considered, because variables created at different iterations could have the same meaning.

To select the most promising literal from the candidates generated at each step, the algorithm considers the performance of the rule over the instances stored in the ontology. The evaluation function used to estimate the utility of adding a new literal is based on the number of positive and negative bindings covered before and after adding this new literal. Let us consider a rule R_i and a candidate literal L_{i+1} that might be added to the body of the rule. The evaluation function is defined as

$$Rule\_Gain(L_{i+1}, R_i) \equiv t\left(\log_2 \frac{p_1}{p_1 + n_1} - \log_2 \frac{p_0}{p_0 + n_0}\right)$$

where p_0 and n_0 are the number of positive and negative bindings of R_i, while p_1 and n_1 are the number of positive and negative bindings of the new rule R_{i+1} (resulting from the addition of L_{i+1}). Finally, t is the number of positive rule R bindings that are still covered after adding literal L_{i+1} to R_i.

The performance of composite concept annotation is tightly related to the reliability of the semantic classifiers. This performance can be improved considering the probability of contemporary presence of individual concept pairs, as well as their temporal consistency. To this end, we included in the ontology the relation of concepts co-occurrence, expressed using mutual information that measures the dependence of a concept pair. This quantity is computed from the analysis of the concept instances as:

$$MI(C_i, C_j) = \sum_{k,l \in \{0,1\}} P(C_i = k, C_j = l) \frac{P(C_i = k, C_j = l)}{P(C_i = k)P(C_j = l)}$$

```
Pos ← Positive examples
Neg ← Negative examples
Rule ← Initial rule
repeat
   Candidate literals ← Generating hypothesis candidates
   Best literal ← argmax_L Rule Gain(L,Rule)
   Add Best_literal to Rule preconditions
   Pos ← subset of Positive examples that satisfy Rule
   Neg ← subset of Negative examples that does not satisfy Rule
until Neg is empty or no more Neg examples are excluded for ι iterations
```

Figure 1. First-order inductive learner for SWRL (Foils) algorithm.

where $MI(C_i, C_j)$ is the mutual information between concept C_i and C_j. The value of $P(C_i = k)$ for $k \in \{0, 1\}$ is the probability of the presence or absence of C_i in the videos. The probability values $P(C_i = k)$, $P(C_j = l)$, and $P(C_i = k, C_j = l$ for $k,l \in \{0, 1\}$ are computed from ground truth. Following the approach introduced elsewhere,[6] it's possible to exploit the mutual information to refine the confidence values of the detected concept instances. Given $P_i = P(C_i = 1|S)$ the confidence score of a detector for the concept C_i in a video shot S and $P = [P_1, \ldots, P_n]^T$ the confidence score vector for all concepts in S, it's possible to refine the confidence scores P^+ with

$$P^+ = (1 - \alpha)P + \alpha MP \quad (1)$$

where $\alpha \in [0, 1]$ weights the contribution of the mutual information and M is a matrix, whose entries have been computed using the mutual information, with the diagonal elements set to 0 to avoid self-reinforcement.

The confidence score of a detector for concept C can be improved considering the fact that the presence of a semantic concept generally spans multiple consecutive shots.[7] In particular, for each concept we reevaluate its confidence values at each shot using:

$$P^T(C^t = 1|S^t) = \sum_{i=-d}^{d} \omega_i P(C^t = 1|C^{t-i} = 1)P(C^{t-i} = 1|S^{t-i}) \quad (2)$$

where $P(C^t = 1|C^{t-i} = 1)$ are probabilities estimated from ground-truth annotations, $P(C^t = 1|S)$ is the confidence score of a detector for the concept C in shot S^t, ω_i is a concept-dependent weighting coefficient (with $\Sigma_i \omega_i = 1$) that measures the contribution from the shot that is temporally i shots apart from S^t, while d is the maximum temporal distance within which the shots are considered.

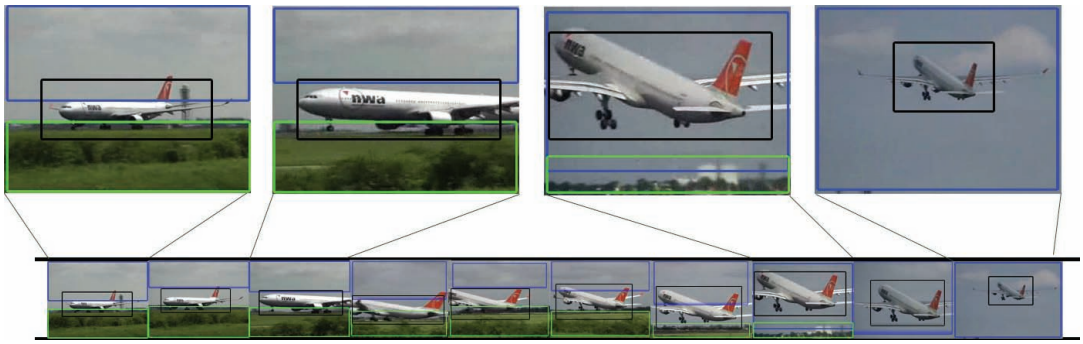


Figure 2. Examples of "airplane," "sky," and "ground" detection and tracking in a Trecvid video sequence.

As an example of rule learning using the Foils algorithm, consider the event, "airplane take-off," and a simple initial rule, such as

$$Airplane(?a) \wedge Sky(?s) \wedge Ground(?g) \rightarrow AirplaneIsTakingOff(?a)$$

The algorithm enriches an initial rule with spatiotemporal relations, using a training set. The literal candidates considered by the algorithm are all the classes and properties defined in the ontology domain (for example, *HasBoundingBox*(?*s*, ?*sBox*), the temporal properties used to encode Allen's logic (for example, *Temporal: before* (?*a*, ?*s*)) and the spatial properties used to encode the relative positions between concepts (for example, *Spatial* : *BoxOverlapsBox*(?*tas*, ?*aBox*, ?*sBox*)). At each step, the most promising literal is added, considering the performance of the rules over the training data, until the recognition performance does not improve. Thus, the result of the Foils algorithm is

$$\begin{aligned} &Airplane(?a) \wedge Sky(?s) \wedge Ground(?g) \wedge \\ &HasBoundingBox(?a, ?aBox) \wedge \\ &HasBoundingBox(?s, ?sBox) \wedge \\ &HasBoundingBox(?g, ?gBox) \wedge \\ &Spatial : BoxOverlapsBox(?tas, ?aBox, ?sBox) \wedge \\ &Spatial : BoxIsInBox(?tag, ?aBox, ?gBox) \wedge \\ &Temporal : After(?tas, ?tag) \wedge MovingObject(?a) \\ &\qquad \rightarrow AirplaneIsTakingOff(?a) \end{aligned}$$

This rule can be translated in the following sentence: IF "airplane," "sky," and "ground" instances (*a*, *s*, *g*) occur in a shot AND they have a bounding box (*aBox*, *sBox*, *gBox*) AND for a time interval *tas*, the bounding box of the airplane is on the bounding box of the sky AND for a time interval *tag* the bounding box of the airplane is on the bounding box of the ground AND the time interval *tas* is after of the interval *tag* AND the airplane is a moving object, THEN that airplane is "taking off." In some cases, we can observe that Foils adds some literals that are not necessary for the event representation. However, this doesn't negatively affect the performance of the rule. In this example, the moving-object concept, that in our ontology is a hypernym of "airplane," is added to the rule even if it is not necessary.

Once the rule is learned, it is applied to the ontology, which contains the instances obtained by the semantic classifiers, to automatically extend the video annotation with instances of the airplane take-off event. In this case, the ontology contains instances resulting from the detection of "airplane," "sky," and "ground" detectors. These detectors have been created using the Viola and Jones algorithm (provided by OpenCV) and color-based pixel classification with a support vector machine, to detect and localize objects. Then, the spatiotemporal evolution of the appearance of concepts is determined using a tracker, based on an improved version of the particle filter.[8] Concept instances are associated with color and luminance histograms, which are used by the tracker to identify each instance in a video sequence. As an example, Figure 2 shows a sequence of "airplane take-off" with results of concept detectors.

Experimental results

We evaluated how much our method improves the performance of individual concept detectors, exploiting concept co-occurrence and temporal consistency. We built an ontology from the MediaMill detectors thesaurus,[9] following the method described. The dataset used for this experiment is the training set of Trecvid 2005; it was divided using a four-fold approach, maintaining groups of consecutive shots in the same fold, to be able to evaluate the effects of time consistency. The parameters of Equations 1 and 2 (α and d, respectively) were chosen in preliminary

experiments on the training data. In the training phase, we identified the concepts that took advantage of the use of the co-occurrence relation, and computed the refined confidence score only for them (based on Equation 1, with $\alpha = 0.1$). The mean average precision (MAP) computed for all the concepts improved by 4.37 percent.

After the co-occurrence refinement, we computed the temporal consistency refinement for all concepts (setting $d = 15$ in Equation 2). The overall improvement of MAP, obtained by the combination of the two techniques, is 17.64 percent. Table 1 shows the performance of the baseline detectors and the results of the two refinement techniques in terms of average precision. We report only the 50 concepts that obtained the largest variations. The concepts whose detectors have a low performance, like ''airplane'', ''desert,'' ''explosion,'' and ''people marching,'' are improved by use of co-occurrence that exploits the results of more robust detectors.

The use of temporal consistency greatly improves the performance of certain concepts that are related to topics often shown in consecutive shots within news videos, like politics (for example, ''Arafat,'' ''Bush Jr,'' and ''government leader'') or sports (for example, ''soccer,'' ''basketball,'' and ''boat''). Small improvements are obtained for detectors with high performance, such as ''anchor,'' ''people,'' and ''outdoor.''

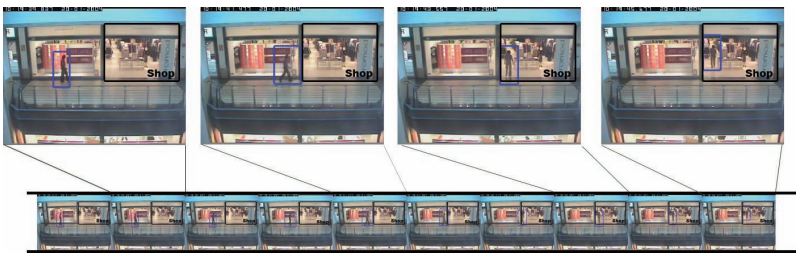
We checked the capability of our system to detect composite concepts using semantic rules, automatically learned from concept instances, in two video domains: broadcast news and surveillance. For the first domain, we considered four events selected from the Large Scale Concept Ontology for Multimedia (LSCOM) events and activities:[10] ''airplane flying,'' ''airplane take-off,'' ''airplane landing,'' and ''airplane taxiing.'' The other set of events is related to the video surveillance of shopping malls: ''person enters a shop'' and ''person exits a shop.'' The dataset used for the news domain consists of 65 Trecvid 2005 videos and 100 videos containing airplane events taken from YouTube, Alice Video (see http://dailymotion.alice.it), PlanesTV (see http://www.planestv.com/planestv.html), and Yahoo! video. This set is available from http://www.micc.unifi.it/dome. We refer to this set in the following as the Web dataset.

Table 1. Average precision of 50 concepts selected from the 101 MediaMill thesaurus, showing comparison of baseline with the proposed refinement approaches: co-occurrence only and the combination of temporal consistency with co-occurrence. The overall improvement for all the concepts, using co-occurrence and temporal consistency, is 17.64 percent.

Concept	Baseline	Co-occurrence	Co-occurrence + temporal consistency
Airplane	0.04	0.08	0.09
Anchor	0.82	0.84	0.84
Animal	0.46	0.46	0.41
Arafat	0.00	0.00	0.14
Basketball	0.35	0.35	0.48
Bird	0.60	0.60	0.62
Boat	0.07	0.07	0.19
Building	0.28	0.29	0.29
Bus	0.01	0.01	0.06
Bush Jr	0.07	0.06	0.30
Car	0.16	0.16	0.18
Cartoon	0.20	0.20	0.25
Chair	0.48	0.49	0.49
Cloud	0.12	0.12	0.47
Desert	0.01	0.06	0.20
Entertainment	0.14	0.14	0.16
Explosion	0.01	0.06	0.21
Female	0.08	0.08	0.11
Fire weapon	0.05	0.12	0.05
Food	0.60	0.60	0.77
Golf	0.21	0.21	0.19
Government leader	0.27	0.27	0.41
Grass	0.05	0.05	0.05
House	0.02	0.02	0.02
Indoor	0.62	0.62	0.60

The Trecvid videos were selected from the Trecvid development set, considering those containing the LSCOM concepts ''airplane take-off,'' ''airplane landing,'' and ''airplane flying.'' We inspected all the videos annotated with the ''airplane'' concept to select those that contain the ''airplane taxiing'' event because this concept is not used in LSCOM. The videos used for the second domain are the Context Aware Vision using Image-based Active Recognition (Caviar; see http://homepages.inf.ed.ac.uk/rbf/CaviarDATA1/) surveillance videos, selected from the front view of the second set. These videos were filmed from a fixed-position camera that frames a mall shop and the area in front of the shop. In the experiments, the scene framed was divided into four parts, as shown in Figure 3 (next page), to determine when a person is in the shop, in front of it,

(a)



(b)

Figure 3. (a) Caviar surveillance video dataset: view of the mall shop areas. (b) Example of person detector and tracking in a video sequence.

Table 2. Precision and recall of actions and events for different datasets.

Data set	Action/event	Precision	Recall
Trecvid 2005	Airplane flying	0.94	0.52
Trecvid 2005	Airplane take-off	0.32	0.40
Trecvid 2005	Airplane landing	0.69	0.69
Trecvid 2005	Airplane taxiing	0.92	0.78
Web	Airplane flying	0.93	0.92
Web	Airplane take-off	0.78	0.81
Web	Airplane landing	0.84	0.94
Web	Airplane taxiing	0.96	0.78
Web + Trecvid 2005	Airplane flying	0.93	0.72
Web + Trecvid 2005	Airplane take-off	0.55	0.60
Web + Trecvid 2005	Airplane landing	0.76	0.81
Web + Trecvid 2005	Airplane taxiing	0.94	0.78
Caviar	Person enters the shop	0.96	0.76
Caviar	Person leaves the shop	0.95	0.89

or in front of the showcase. The two datasets were divided using a three-fold approach, to learn the rules.

We used these rules to annotate the videos, evaluating the results in terms of precision and recall, as shown in Table 2. The overall results for all the rules are extremely promising. The performance of ''airplane flying'' and ''airplane taxiing'' is better than that of ''airplane landing'' and ''airplane take-off.'' This is due to the fact that the rules modeling those events are simpler.

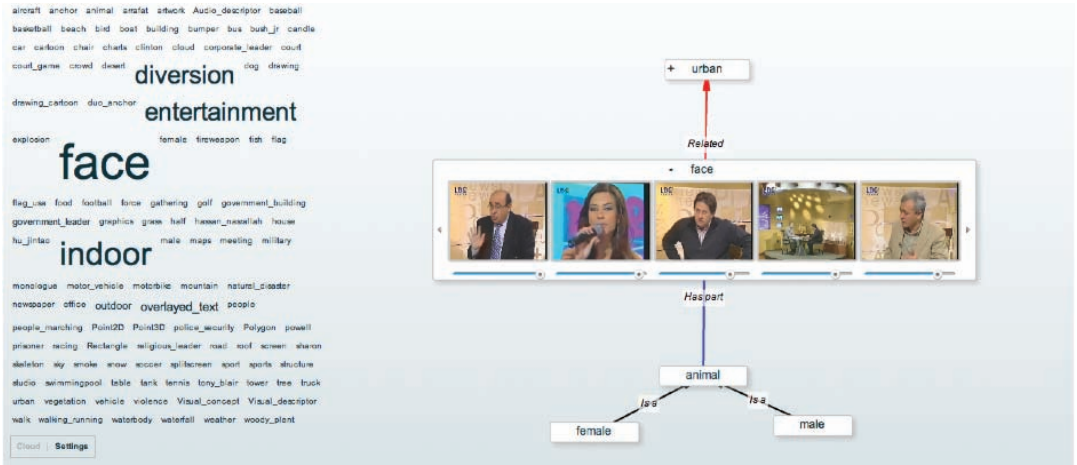
The performance of the rules depends on the performance of the detectors and tracker. Investigation of the cases in which the rules fail has shown that the main cause of failure is due to the performance of the sky and ground detectors. In particular, these detectors are affected by the low quality of the images and the presence of superimposed graphics. In a few cases, the fault was the airplane detector, especially when superimposed graphics and text covered the appearance of the airplane, which occurred mostly in Trecvid videos. This fact is reflected by the different performance in the two datasets.

The results of the recognition of video-surveillance actions show good performance in precision and recall. The fixed camera and lighting conditions reduce the variability of the appearance of the observed events and objects, leading to good performance from the person detector and the tracker. The performance of the rules mainly depends on the errors of the tracker, which sometimes happened with multiple persons' trajectories overlapping.

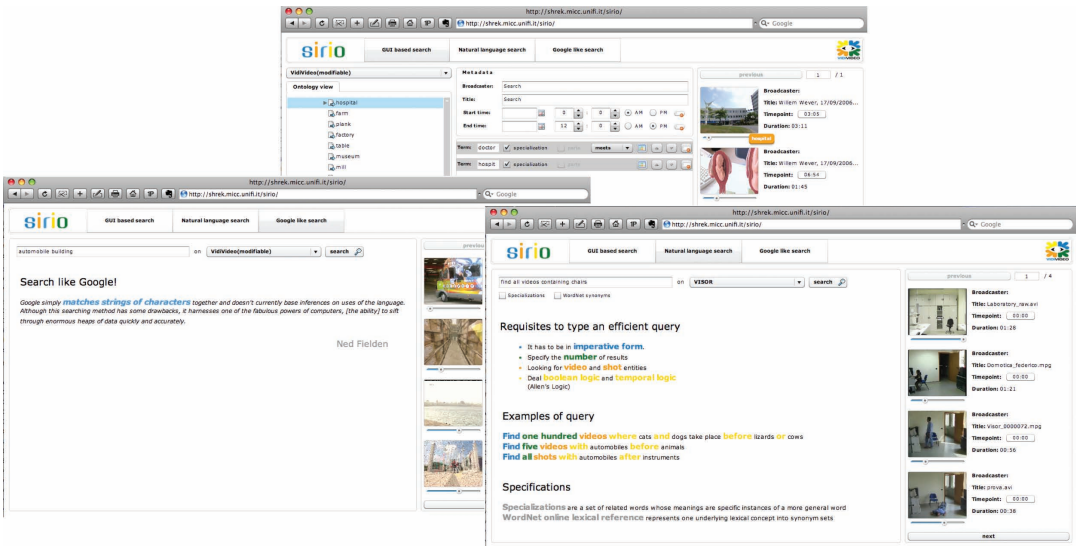
Sirio search engine

Browsing and searching video archives is performed by exploiting the ontology with a Web-based prototype system called Sirio (see http://www.micc.unifi.it/vidivideo; contact article authors to obtain access passwords), which provides integrated support for Boolean-temporal, semantic, and query-by-example queries. The system is based on the rich-Internet-application paradigm. Rich Internet applications can avoid the usual slow and synchronous loop for user interactions, typical of Web environments that use only the HTML widgets available in standard browsers. This has allowed us to implement a visual-query mechanism that exhibits a look and feel approaching that of a desktop environment, with the fast response that is expected by users.

The search engine is a Web application written in Java and executed in an Apache Tomcat application server; supporting multiple ontologies (for different video domains), ontology reasoning services, and W3C SPARQL Protocol and



(a)



(b)

Figure 4.
(a) Andromeda browsing interface: the ontology graph view is used to explore parts of the full ontology, checking the instances of video clips annotated with the selected concept. All the instances of a concept are visible as streaming video clips. (b) Sirio search interfaces: GUI query builder, natural language search, and Google-like search.

RDF Query Language (SPARQL) queries. The GUI is a Flash application, written in Flex and executed in a client-side Flash Virtual Machine. Videos, returned as query results, are streamed using the Real-Time Messaging Protocol. To browse an archive, inspecting the annotated concept instances (that is, video clips), the user navigates the ontology structure, presented as a graph. Figure 4a shows the browser interface. The user can select a concept from a tag cloud that shows the concepts with the largest number of instances or navigate the ontology following the concept relations.

The prototype provides different search modalities, as shown in Figure 4b, designed for different types of users. There is a GUI to build composite queries that include Boolean-temporal operators (based on Allen's logic), visual prototypes for query-by-example, and video metadata (such as broadcaster and program names, broadcast dates, and so on) for professional users. A free-text interface for Google-like searches and a natural language interface lets users compose queries with Boolean-temporal operators. These approaches are suitable for novice users because they don't require them to specify complex queries or broadcast metadata. Using the ontology relations and reasoning, it's possible to extend user queries through subsumption and meronymy. The natural-language and Google-like interface require another form of query expansion, using synonym relations based on WordNet, so users can formulate their queries naturally, without being forced to select terms from a lexicon.

Conclusions

Field trials and tests with professional archivists of the search and browsing engine have shown the potential of the use of ontologies for annotation and retrieval. Our future work will deal with the learning of rules that cope with uncertainty and using fuzzy ontology reasoning that can exploit detector confidence scores. In addition, we plan to investigate the use of fuzzy temporal Horn logic to overcome the expressivity limitations of SWRL. **MM**

Acknowledgment

This work is partially supported by the European Information Society Technologies Vidi-Video Project (contract FP6-045547) and the IM3I Project (contract FP7-222267).

References

1. M. Everingham et al., ''The PASCAL Visual Object Classes Challenge 2009 (VOC) Results,'' 2009; http://www.pascal-network.org/challenges/VOC/voc2009/workshop/index.html.
2. A.F. Smeaton, P. Over, and W. Kraai, *High-Level Feature Detection from Video in Trecvid: A 5-Year Retrospective of Achievements,* Springer Verlag, 2009.
3. A. Hauptmann et al., ''Can High-Level Concepts Fill the Semantic Gap in Video Retrieval? A Case Study with Broadcast News,'' *IEEE Trans. Multimedia,* vol. 9, no. 5, 2007, pp. 958-966.
4. J. Yang and A. Hauptmann, ''(Un)reliability of Video Concept Detection,'' *Proc. ACM Int'l Conf. Image and Video Retrieval,* ACM Press, 2008, pp. 85-94.
5. J.R. Quinlan, ''Learning Logical Definitions from Relations,'' *Machine Learning,* vol. 5, no. 3, 1990, pp. 239-266.
6. Z.-J. Zha et al., ''Building a Comprehensive Ontology to Refine Video Concept Detection,'' *Proc. ACM Int'l Workshop Multimedia Information Retrieval,* ACM Press, 2007, pp. 227-236.
7. K.-H. Liu et al., ''Association and Temporal Rule Mining for Post-Filtering of Semantic Concept Detection in Video,'' *IEEE Trans. Multimedia,* vol. 10, no. 2, 2008, pp. 240-251.
8. A.D. Bagdanov et al., ''Improving the Robustness of Particle Filter-Based Visual Trackers Using Online Parameter Adaptation,'' *Proc. IEEE Int'l Conf. Advanced Video and Signal Based Surveillance,* IEEE Press, 2007, pp. 218-223.
9. C. Snoek et al., ''The Challenge Problem for Automated Detection of 101 Semantic Concepts in Multimedia,'' *Proc. ACM Multimedia,* ACM Press, 2006, pp. 421-430.
10. L. Kennedy, *Revision of LSCOM Event/Activity Annotations, DTO Challenge Workshop on Large Scale Concept Ontology for Multimedia,* Advent technical report #221-2006-7, Columbia Univ., 2006.

Lamberto Ballan is a PhD student at the Visual Information and Media Lab at the Media Integration and Communication Center, University of Florence, Italy. His research interests include multimedia information retrieval, pattern recognition, computer vision, and machine learning. Ballan has a laurea degree in computer engineering from the University of Florence. Contact him at ballan@dsi.unifi.it.

Marco Bertini is an assistant professor in the Department of Systems and Informatics at the University of Florence, Italy. His research interests include content-based indexing and retrieval of videos and Semantic Web technologies. Bertini has a PhD in electronic engineering from the University of Florence. Contact him at bertini@dsi.unifi.it.

Alberto Del Bimbo is a full professor of computer engineering at the University of Florence, Italy, where he is also the director of the master in Multimedia Content Design. His research interests include pattern recognition, multimedia databases, and human–computer interaction. Del Bimbo has a laurea degree in electronic engineering from the University of Florence. Contact him at delbimbo@dsi.unifi.it.

Giuseppe Serra is a PhD student at the Visual Information and Media Lab at the Media Integration and Communication Center, University of Florence, Italy. His research interests include multiple-view geometry, self-calibration and 3D reconstruction, and video understanding based on statistical pattern recognition and ontologies. Serra has a laurea degree in computer engineering from the University of Florence. Contact him at serra@dsi.unifi.it.