Aggregated Feature Retrieval for MPEG-7

Jiamin Ye[1] and Alan F. Smeaton

Centre for Digital Video Processing, Dublin City University, Glasnevin, Dublin 9, Ireland.
{jiaminye, asmeaton}@computing.dcu.ie

Abstract. In this paper we present an initial study on the use of both high and low level MPEG-7 descriptions for video retrieval. A brief survey of current XML indexing techniques shows that an IR-based retrieval method provides a better foundation for retrieval as it satisfies important retrieval criteria such as content ranking and approximate matching. An aggregation technique for XML document retrieval is adapted to an MPEG-7 indexing structure by assigning semantic meanings to various audio/visual features and this is presented here.

1. Introduction

The challenge for video search is to provide support for the following capabilities:

- Textual queries at different segment levels such as individual frames, shots, scenes, or perhaps whole programmes. Available information for searching should include meta-data, teletext and ASR transcripts.
- Image or video clip queries also at different segment levels. Available support for searching should include various visual and audio features.
- Content-based indexing and similarity comparisons. A video IR system needs to utilise available audio-visual content-based indexing and retrieval techniques. Visual indexing should include techniques like region colour, edge component histogram and face recognition. Audio indexing should use techniques like speech and music recognition. Similarity comparison techniques will vary depending on the indexing technique used.

The above list is representative of what contemporary video IR demands and building systems which realise this means engineering complex systems where interoperability is also desirable and the MPEG-7 video standard delivers an appropriate framework for building this type of video search engine. MPEG-7 is a recently developed international standard for multimedia mark-up where the language describes the syntax and semantics to describe features of multimedia contents at various granularity levels [1]. The challenge for MPEG-7 is supporting video retrieval, allowing a variety features to be indexed on which retrieval can be based.

[1] This paper is based on research undertaken by the first author as a Ph.D.student.

In this paper we present an initial study of the use of both high and low level MPEG-7 descriptions for video retrieval. Some related work based video retrieval for MPEG-7 will be given in section 2. Since an MPEG -7 description is a standardised XML document, a short survey on current XML indexing techniques is given in section 3 to address their strengths in terms of video retrieval. This shows that IR-based XML indexing provides a better foundation since it satisfies criteria such as content ranking and approximate matching, particularly useful in schema-constrained environments. Section 4 explains the aggregation technique used in IR-based XML indexing. An MPEG-7 indexing framework for video retrieval is given in section 5. This extends the Kazai et al's [2] model to handle audio/visual features as auxiliary evidence to the existing term weight indexing. Finally, we then conclude the paper.

2. Related Work Based on Video Retrieval for MPEG-7

There are a number of related projects that involve searching MPEG-7 descriptions but two are of particular interest. The SAMBITS project has built a consumer terminal that supports the MPEG-7 storage and delivery [3]. Queries to this system are text based and consider content (e.g. keywords) and meta-data (e.g. authors), possibly with an XML structure. The advantage is the ability to integrate content-based, fact-based and structural information.

The second project has designed an MPEG-7 retrieval model based on inference networks and was developed to capture the structural, content-based and positional information by a Document and Query network [4]. Positional information is context-related and contains and the location of content within a document. The MPEG-7 inference network has three layers: Document, Contextual and Conceptual while the Query network consists of concept and context nodes and query operators. Retrieval is done in a bottom-up fashion by accumulating evidence from all layers to find the best shots.

3. Survey of XML Indexing Techniques

Some insights into current XML indexing and retrieval methods are now given as these methods can be tailored for MPEG-7 video retrieval. The methods can be classified into three categories:

- *Information retrieval based indexing* brings the structure of documents together with the content into one unified model. Text indexing methods generally use the traditional tf-idf term weight indexing structure and its variants. Returned items are ranked based on their estimated relevance to a given query and most studies emphasise developing the right ranking formulae such as aggregation-based retrieval within schema-constrained environments [2].
- *Path expression based indexing* regards an XML document as a rooted ordered graph containing a set of nodes (i.e. element ID) and a set of edges (i.e. element name or attribute name). The path index such as Dataguides [5] maintains the

structural summaries of an XML collection. Each entry of the summary is a sequence of edges and attached to a target set of element IDs that are the last node from some path instances in the graph which satisfy the entry. The type of queries used is based on XML structure conditioned on search terms or values. Returned results are the path expression of XML structure rather than the content.

- *Tree matching based indexing* determines whether the query tree approximately appears in the XML data tree and sorts results based on the score of the XML structure rather than content since no term index is provided. *Approximate* matching is measured by the number of paths in the query tree that match against the data tree [6].

The requirements for indexing and retrieval using MPEG-7 are for ranked video segments, textual queries and approximate matching. Three possible types of XML indexing techniques are compared and it shows that the IR-based indexing technique provides a better foundation for multimedia MPEG-7 retrieval since it satisfies most of the listed criteria and is particularly useful in a schema-constrained environment.

4. Aggregation of Term Weights

The aggregation-based XML indexing structure is represented as a graph whose nodes are elements within a document hierarchy and whose edges reflect structural relationships between the connected nodes [2]. Three types of relationships are of interest: hierarchical (parent-child), linear (siblings) and referential (hyperlink). A document component C is not only associated with concept A_j (1≤j≤n) within its own content A_j^C but also the content A_j^{si} of its structurally related components S_i (1≤i≤k). The assumption of document components being about concepts is interpreted by those containing terms that make a good concept as indicated by $t(A_j^C)$. The aggregation of an overall score for C combines the weights in content A_j^C and A_j^{si} based on three factors: the type of structural relationships, the type of content and the linguistic quantifiers used. The basic model of aggregation is introduced below namely OWA operators and linguistic quantifiers.

4.1 OWA Operators in Structured Document Retrieval

Given a document node C, we have the weight t_0 of a term in its own content A_0^C and the weights t_j in its k structurally related contents A_0^{si} (1≤j≤k). Vector $t = \{t_0 ,\dots t_k\}$ is the *argument vector* of the term and vector w = $\{w_0 ,\dots w_k\}$ indicates the importance value associated with the corresponding term weight of component. An *Ordered Weighted Averaging* (OWA) operation with respect to t is obtained as follows:

1. Sort entries of vector t in descendent order and obtain the *ordered argument vector* $b = \{ b_0 ,\dots b_k \}$, where b_j is the j-th largest of t_i. The ordering of the corresponding importance weighting vector w is changed to that of vector b, denoted as $\alpha = \{ \alpha_0,\dots \alpha_k \}$ where α_j is the importance value of the j-th largest of t_i.

2. Apply the OWA operator as follows to obtain the overall score t^C of component C:

$$F(t^C) = \sum_{j=0}^{k} \alpha_j b_j \quad where \quad \alpha_j \in [0,1] \quad and \quad \sum_{j=0}^{k} \alpha_j = 1 \tag{1}$$

4.2 Linguistic Quantifiers

Having obtained the argument vector *t*, the task left in aggregation is to determine a suitable importance weighting vector *w*. Words such as *"most"*, *"about ¼"* and *"at least one"* are called *linguistic quantifiers* describing a proportion of items in a collection. Each quantifier is related to a regularly increasing monotonic function Q to translate natural language terms into a measurable value w_j[7]. Examples of Q are Q = r^p where p ranges [0,∞), when p = 2 the function implies quantifier *most* which places most of the importance on the lower weights b_j thereby emphasising the lower scores.

In automatic document retrieval, it is possible to determine the ordered importance weighting vector α by some measurable property of the document such as term weights t_j. Each entry of α_j can be calculated based on the formula below to achieve $\alpha_j \in [0,1]$ and $\Sigma_j \alpha_j = 1$:

$$\alpha_j = Q(S_j) - Q(S_{j-1}) \quad where \quad S_j = \frac{\sum_{j=0}^{j} b_j}{T} \quad and \quad T = \sum_{j=0}^{k} b_j \tag{2}$$

T is the sum of all ordered argument weights b_j of a term and S_j is the sum of *j* ordered argument weights of the j-th most satisfied components.

5. An Indexing Framework for MPEG-7 Video Retrieval

An MPEG-7 search engine which such as outlined earlier attempts to retrieve video at the shot level limiting its use to high-level evidence such as meta-data and manual text annotation. The goal of this paper is to study the possibility and feasibility of bringing low-level visual features as auxiliary evidence to improve the retrieval results. In trying to achieve this the aggregation technique shows its ability to model textual representations of a component and that of its structurally related components. Our solution is to map a visual feature of a given shot into a term weight vector, or possibly a concept vector based on the assumption that shots in the same cluster present similar visual/audio features and are designed to capture a similar semantic meaning. The dialogue (e.g. ASR transcripts or teletext) of the shots in the cluster will most likely exhibit similar patterns of term occurrences. Linking audio/visual features to term weights can provide auxiliary evidence to the existing shot dialogue. A crucial concern is that the language variations used in the dialogue and pure term occurrence patterns may not be enough. Our future work will attempt to map audio/visual features to a higher-level concept representation via the characteristics of dialogue term usage.

5.1 Cluster Semantic Assignment: Term-by-Cluster Matrix

Clustering is intended to identify structures hidden in data sets using an unsupervised learning approach. A k-means clustering technique can be applied to shots for each audio/ visual feature [8] and having obtained shot clusters, we can estimate the extent $P(C_k|S_j)$ to which a given shot S_j falls into cluster C_k by Eq (3):

$$P(C_k \mid S_j) = \frac{P(C_k)P(S_j \mid C_k)}{\sum_{k=1}^{n} P(C_k)P(S_j \mid C_k)} \tag{3}$$

where $P(S_j|C_k)$ is the distance measure between shot S_j and the centre of cluster C_k., $P(C_k)$is the accuracy of cluster C_k being correctly classified, n is the total number of clusters, and $P(C_k|S_j)$ is the probability that a given shot belongs to a cluster.

Cluster semantic assignment is used to connect the feature cluster with a semantic representation, and as previously mentioned this could be a term weight vector. A term-by-cluster matrix, similar to term-by-document matrix, is used to record the degree of association between any clusters and terms in the MPEG-7 collection including titles, program abstract and dialogues. This matrix has k columns and m rows, where k is the total number of clusters and m is the total number of indexed terms (Figure 1).

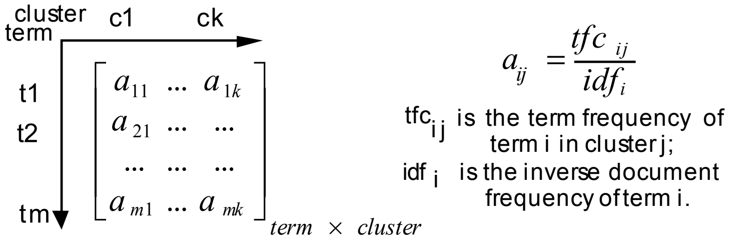
cluster
term
c1
ck
t1
t2
tm
a_{11} ... a_{1k}
a_{21} ... ...
... ...
a_{m1} ... a_{mk}
term × cluster
$a_{ij} = \frac{tfc_{ij}}{idf_i}$
tfc_{ij} is the term frequency of term i in cluster j;
idf_i is the inverse document frequency of term i.

Fig. 1. The Term-by-cluster matrix A

A Singular Value Decomposition technique can be used to derive semantics from matrix A. SVD was originally developed for text retrieval to provide term similarity information based on document co-occurrences [9]. The same can be applied to the term-by-cluster matrix, decomposing A into a diagonal matrix and two orthogonal matrices to obtain an approximation by choosing the number of singular values j:

$$A'_{term \times cluster} = U_{term \times j} S_{j \times j} V_{j \times cluster}^{T} \tag{4}$$

where S is the diagonal matrix with only the first k non-zero singular values in descending order. U and V are matrices consisting of the first k columns corresponding to the singular values in S and each column is with unit-length.

We can have the following by observing the matrix A':

- A *Term-cluster* comparison is the individual cell of matrix A': a'_{ij} .
- A *Cluster-cluster* comparison is the dot product of two vector rows of matrix A'.

- A *Term-term* comparison is the dot product of two vector columns of matrix *A'*.

5.2 The Overall Structure

An MPEG-7 document can be regarded as a tree whose nodes are the *retrievable* points from which users can then start video browsing or playback and whose edges represent hierarchical relationships (Figure 2). Each node has an attribute represented by a term weight vector based on the content, which can be directly obtained from the MPEG-7 documents such as a document abstract and ASR transcripts. There exists special types of attributes that are calculated from a term-cluster mapping function which we call *auxiliary attributes* to shots, namely they are various visual features originally indexed by pre-computed feature vectors.

The video indexing task is to first obtain the overall representation of a shot by combining the weight of Text-Annotation and those of auxiliary attributes. We can then use the aggregation method to compute the overall representation of the non-leaf nodes. The completed index structure is shown in Figure 3. Retrieval can be done in a straightforward manner by converting any given query into a term weight vector, obtaining the dot products between query vector and the overall term weight vector of all the nodes in the MPEG-7 tree, and finally sorting the dot product values in descending order for result display.

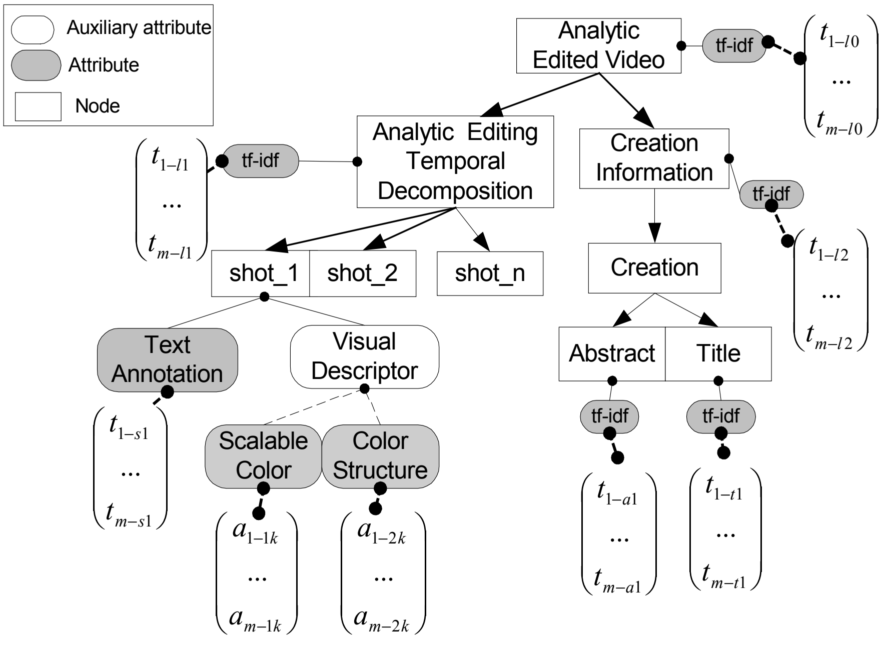
Auxiliary attribute
Attribute
Node
Analytic Edited Video
tf-idf
t_{1-l0} ... t_{m-l0}
Analytic Editing Temporal Decomposition
tf-idf
t_{1-l1} ... t_{m-l1}
Creation Information
tf-idf
t_{1-l2} ... t_{m-l2}
shot_1
shot_2
shot_n
Creation
Text Annotation
t_{1-s1} ... t_{m-s1}
Visual Descriptor
Scalable Color
a_{1-1k} ... a_{m-1k}
Color Structure
a_{1-2k} ... a_{m-2k}
Abstract
tf-idf
t_{1-a1} ... t_{m-a1}
Title
tf-idf
t_{1-t1} ... t_{m-t1}

Fig. 2. Before indexing MPEG-7, a tree structure representation

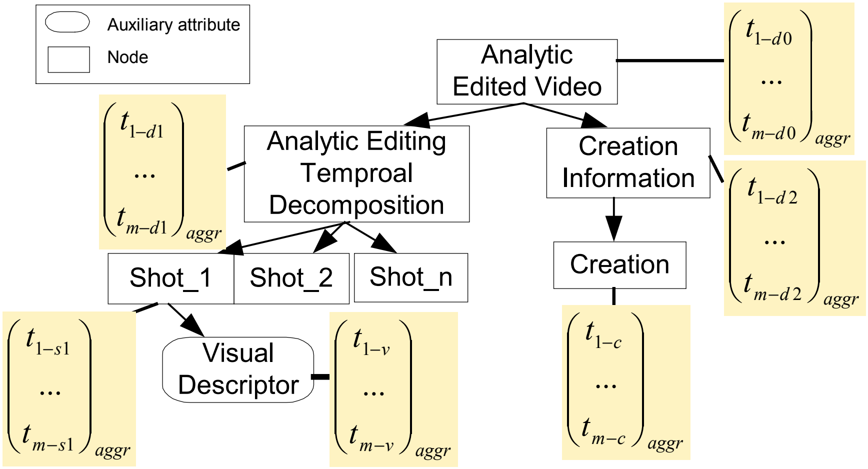
Auxiliary attribute
Node
Analytic Edited Video
Analytic Editing Temproal Decomposition
Creation Information
Creation
Shot_1
Shot_2
Shot_n
Visual Descriptor
t_{1-d0} … t_{m-d0} aggr
t_{1-d1} … t_{m-d1} aggr
t_{1-d2} … t_{m-d2} aggr
t_{1-s1} … t_{m-s1} aggr
t_{1-v} … t_{m-v} aggr
t_{1-c} … t_{m-c} aggr

Fig. 3. After indexing MPEG-7 : each node has its corresponding aggregated term weight vector, as indicated with light yellow background.

5.3 Combining Weights of Auxiliary Attributes

Aggregation can be used in situations when there exists a *direct* dependence relationship between an attribute and a node such as the *Text-Annotation* attribute and *Shot* node. It is not suitable for combining weights of an *auxiliary* attribute such as visual features because the relationship between a shot and a term is conditioned upon the shot's feature clusters. A simplified Bayesian approach can be used to obtain the belief value $P(t_l|S_l)$ of an index term t_l within a given shot S_l conditioned upon the corresponding visual feature clusters (i.e. the degree of belief that shot S_l supports term t_l via clusters). This helps summarise the visual features based on some concrete textual words rather than the abstract colour or shape information. It is somewhat implicit representing the semantic of the visual features in a quantitative way.

The Bayesian inference net model assumes that random variables take up two values {true, false}. To simplify the computation, we approximate the belief by using only true values. The strength of the relationship is given in principal by a conditional probability. The task is to find the degree of belief $P(t_l \mid S_l)$ shown below:

$$\begin{aligned} P(t_1 \mid s_1) = {} & P(t_1 \mid c_1)P(c_1 \mid s_1)P(s_1) \\ & + P(t_1 \mid c_2)P(c_2 \mid s_1)P(s_1) \\ & + P(t_1 \mid c_3)P(c_3 \mid s_1)P(s_1) \end{aligned} \tag{5}$$

where $P(C_j \mid S_1)$ is the probability that a given shot belongs to a cluster C_j (see Eq.3); $P(t_1|C_j)$ is the probability that that a given feature cluster is implicitly expressed by term t1 (see term-by-cluster matrix in section 5.1).

6. Conclusion

This paper describes a study of an MPEG-7 indexing framework for video retrieval extended from the aggregation based indexing model of structured document retrieval. The main thrust of this paper focuses on mapping visual/audio features into term weight vectors thereby providing auxiliary evidence to the overall representation of a shot. The cluster semantics assignment helps to find semantic meanings (i.e. a term weight vector) for each feature cluster based on the assumption that programmes such as news and documentarys provide a rich set of words and have a consistent and structured language style.

Our future work will implement our framework to evaluate its effectiveness in retrieving video shots based on various query types: textual, image and video clip. To aid evaluation our experiments will be carried out on the TREC2002 video search collection. This search collection consists of 40 hours MPEG-1 video and 25 queries addressing users' information needs in various types. It also provides a set of MPEG -7 descriptions for shot boundary and ASR transcripts. To facilitate clustering assignment, we need to generate low-level visual/audio MPEG-7 descriptions based on our own feature extraction tools since those of TREC2002 only give information on whether a given feature is present in a given shot or not. The experiments will provide us with an insight into retrieval effectiveness by combining visual features in video search.

References

1. Day, N. and Martínez, J.M. (ed.): Introduction to MPEG-7 (v4.0), working document N4675, (2002). Available at: http://mpeg.telecomitalialab.com/working_documents.htm, last visit on 29th Oct, 2002
2. Kazai, G., Lalmas, M. and Rölleke T.: A Model for the Representation and Focussed Retrieval of Structured Documents based on Fuzzy Aggregation. In: String Processing and Information retrieval (SPIRE 2001) Conference, Laguna De San Rafael, Chile. Nov (2001)
3. Pearmain, A., Lalmas, M., Moutogianni, E., Papworth, D., Healy, P. and Roelleke T.: Using MPEG-7 at the Consumer Terminal in Broadcasting. In: EURASIP Journal on Applied Signal Processing, Vol.2002, No.4, April 2002, 354-361
4. Graves, A. and Lalmas, M.: Video Retrieval using an MPEG-7 Based Inference Network. In: Proceedings of SIGIR'02, Tampere, Finland, August 2002, 339-346
5. McHugh, J., Widom, J., Abiteboul, S., Luo, Q. and Rajaraman, A.: Indexing Semistructured Data. Technical Report, January, (1998)
6. Chen, Z., Jagadish, H.V., Korn, F., Koudas, N., Muthukrishnan, S., Raymond Ng, and Srivastava, D.: Counting twig matches in a tree. In: Proceedings of IEEE International Conference on Data Engineering, Heidelberg, Germany, April 2001. 595--604
7. Yager, R.: A Hierarchical Document Retrieval Language. Information Retrieval, Vol.3, No.4, December (2000), 357-377
8. Buhmann, J.M.: Data clustering and learning. In: Arbib, M. (ed.): Handbook of Brain Theory and Neural Networks. Bradfort Books, MIT Press, (1995)
9. Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R.: Indexing by latent semantic analysis. Journal of the ASIS, 41(6), (1990), 391-407