3K－4

A Uniform Visual-Spatio-Temporal Model for Indexing and Retrieving Videos.

Zaher AGHBARI, Kunihiko KANEKO, and Akifumi MAKINOUCHI
Graduate School of Information Science and Electrical Engineering,
Department of Intelligent Systems, Kyushu University 6-10-1 Hakozaki,
Higashi-ku, Fukuoka-shi 812-8581, Japan
E-mail: {zaher,kaneko,akifumi}@db.is.kyushu-u.ac.jp

Abstract

In this paper, we present a uniform approach to represent videos. The visual features and the spatio-temporal features of *objects* in a video are represented uniformally by a topological model. These uniform visual-spatio-temporal (VST) features are used to represent the video shots. The uniform VST topological representation of objects allowed us to define new relations between objects. These new relations support the formulation of more semantical queries and increase the expressiveness of the *VST* video model. We also proposes an automatic extraction of the VST features. In addition, *VST* video model supports similarity queries.

1 Introduction

Recently, two lines of research investigated indexing and retrieving video data. One line of research, [1] [2], utilized the visual features (color, motion, texture, shape, etc.) to represent, index, and retrieve videos. The other line of research [3][4] exploited the spatio-temporal features of video objects to represent, index, and retrieve videos.

In this paper we propose a video model that integrates the works of both lines, visual and spatio-temporal, in one unified visual-spatio-temporal, *VST*, model. As a result, we defined a rich set of relationships based on the visual and spatio-temporal features. The *VST* features are modeled uniformally by a topological approach. The *VST* video model is designed in an object-oriented approach because of its suitability for multimedia data. Therefore, our video model is based on the representation of the *VST* features of the contents of the *video objects*, which is a semantically meaningful object in the scene.

2 Related Work

The visual features are considered the basis for the abstractions of the video semantic contents. In our previous systems, [1] and [2],we extracted every nth frame of a shot and computed their average frame. The average frame is considered to best represent the shot since the average frame captures the visual temporal changes within the shot. Indexes are computed using the extracted average visual features.

The works in [3] and [4] represent videos by the spatial and temporal relationships between the objects in a scene. Both works, [3][4], are based on Egenhofer's topological relations (*disjoint, contain, inside, meet, equal, covers, covered-by, overlap*). Also, they are based on Allen's interval temporal relations (*before, meets, overlaps, during, starts, finishes, equal, and their inverses*). The work in [3] modeled both topological and directional spatial relationships. It, also, integrated their video model into an ODMG based object database system. The work in [4] also integrated both topological and directional relationships, and presented a retrieval method based on a 2D-Projection Interval Representation (2D-PIR).

3 VST Video Model

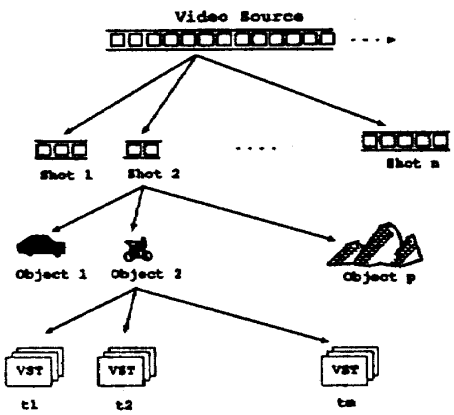

Figure 1: Video model hierarchy

The hierarchical structure of a video can be represented by an acyclic graph as shown in Figure 1. Each video, V, is segmented into a number of *shots* ($S = S_1, S_2, ..., S_n$).A *shot*, S, is a consecutive se-

Table 1: VST Features

Feature	Absolute	inst.	non-inst.
color histogram	Δ	Δ	
motion histogram	Δ	Δ	
position	Δ	Δ	
topological rel.		Δ	
interval temporal rel.		Δ	
relSize rel.		Δ	Δ
relPosition rel.		Δ	Δ
relSpeed rel.		Δ	Δ
relDirection rel.		Δ	Δ
relAppearancetime rel.			Δ

quence of frames that constitute one camera operation, and it is the smallest unit of video data. From each shot, a set of semantically meaningful *video objects* $(O_1, O_2, ..., O_q)$ are extracted. Then, the visual-spatio-temporal, *VST*, features are extracted from each object.

3.1 VST features

The *VST* features of a video object can be divided into two categories based on their extraction time, as shown in Table 1. The *instantaneous* features are extracted at one instant, t_i. The *non-instantaneous* features are extracted from two or more instants, $t_i, t_{i+1}, \ldots, t_n$.

The *VST* video model extracts and stores the absolute instantaneous features, color histogram, motion histogram, and position, of every object at each instant. Other non-absolute features are computed from the stored absolute features when needed at the time of matching.

Since our video model includes both visual and spatio-temporal features of video objects, new relations whose definitions depend on the absolute features, topological relations, and interval temporal relations are introduced. These new relations are: 'relSize', 'relPosition', 'relSpeed', 'relDirection', and 'relAppearaceTime' which define the relative size, relative position, relative speed, relative direction, and relative appearance/disappearance time relationships between video objects, respectively.

The *VST* model presents a uniform topological approach to model the VST features. For every video object in a shot, the absolute features are computed and stored in an object-oriented database. The visual feature are computed as in [1] and [2]. The position of an object is determined by the two points, (x_1, y_1) and (x_2, y_2), that specify the position of its minimum bounding box. Then, the other features, non-absolute, are computed from the stored absolute features when needed during the matching process.

4 Querying

In order to support a similarity retrieval, we assigned values for the degrees of similarity between the different items of colors, directions, and speeds. And, we assigned values for the distances between the different newly defined relationships. Since our video model supports multi-feature, ξ, indexes, a user may formulate a query that investigate several features, $\xi_1, \xi_2, \ldots, \xi_n$. Therefore, a user query is divided into a number of subqueries equal to a number of features, n. Then, each subquery is executed separately. The net distance, D, between a user query and a target shot is equal to the sum of all the distances, d, of the subqueries divided by the number of subqueries, n.

$$D = \frac{\sum_{i=1}^{n} d_i}{n}$$

Query: Find a shot that has two red objects where the smaller one moves faster and disappears before the bigger one.

```
select shot
from   o1, o2 in Objects, shot in VideoShots
where  o1.color(red,100%) and
       o2.color(red,100%) and
       smaller(o1,o2) and faster(o1,o2) and
       outBefor(o1,o2)
```

5 Conclusion

The *VST* video model integrates both visual and spatio-temporal approaches that index and retrieve video data. The uniform modeling allowed us to define a rich set of new relationships between video objects which increased the expressiveness of our video model.

References

[1] Z.Aghbari,K.Kaneko, A.Makinouchi. *New Indexing Method for Content-Based Video Retrieval and Clustering for MPEG Video Database.* Int'l Symposium on DMIB'97, pp.140-149, Nov.1997, Nara, Japan.

[2] Z.Aghbari,K.Kaneko, A.Makinouchi. *A Motion-Location Based Indexing Method for Retrieving MPEG Videos.* to be appeared in the proc. of DEXA'98, Vianna, Austria, Aug. 1998.

[3] J.Z.Li,M.T.Ozsu,D.Szafron. *Modeling of video spatial relationships in an object database management system.* Int'l workshop Multimedia Database Management Systems, pp.124-133, NY, Aug.1996.

[4] M.Nabil, A.H.Ngu, J.Shepherd. *Picture Similarity Retrieval Using the 2D Projection Interval Representation.* IEEE Trans. on Knowledge and Data Engineering. Vol.8(4), pp.533-539. Aug.1996.