A New Video Semantic Model Based on 3D C-string Knowledge Representation

Ping Yu
Dept. of Information Management,Chinese Culture University
yp@faculty.pccu.edu.tw

Anthony J.T. Lee
Dept of Information Management
National Taiwan University
jtlee@ntu.edu.tw

NaiWen Kuo and Chein-Shung Hwang
Dept. of Information Management,Chinese Culture University
neven@faculty.pccu.edu.tw
cshwang@faculty.pccu.edu.tw

Abstract

Modeling object's semantic knowledge has attracted increasing attention in the area of video content management. In this paper, we propose a video semantic model to get the higher-level semantics of spatial relation changes between the objects in a video represented by a 3D C-string, which represents the lower-level information of spatio-temporal relations, motions and size changes of the objects in a video. We use the concept of finite automata to record the transitions of objects' spatial relations. From the final states of the finite automaton, the higher-level semantics of spatial relation changes between the objects in a video can be inferred.

Keywords*: Video semantic model, Video content management, Spatial relation, 3D C-string*

1. Introduction

With the advances in information technology, videos have been promoted as a valuable information resource. However, because of videos' volume and rich content, efficient access to videos is not an easy task. It is required for a video database management system to provide a query with higher-level semantics to meet a user's need. Unlike an image that only contains spatial relations between objects, a video contains richer information such as spatio-temporal relations between objects, the motions and size changes of objects. By analyzing the lower-level information of spatio-temporal relations, motions and size changes of objects, we may obtain the higher-level semantics of spatial relation changes between the objects in a video.

A number of techniques for video content modeling involving temporal events have been proposed. Some of these techniques rely on modeling the interplay among objects over time along with spatial relations between these objects [2]. Hu et al. [3] proposed a cluster-based tracking algorithm to acquire motion trajectories and cluster hierarchically using the spatio-temporal information. From the learning activity model, they construct a hierarchical semantic for indexing and retrieving the objects' activities. The PC-FSM model uses finite automata to analysis video and generates personalized highlights of sport events [1]. For the semantic gap between what we can derive automatically from the visual data and the semantic interpretation a user has of the same data, are discussed in [5].

In this paper, we propose a video semantic model to get the higher-level semantics of spatial relation changes between the objects in a video represented by a 3D C-string [4]. The 3D C-string represents the lower-level information of spatio-temporal relations, motions and size changes of the objects in a video. We use the concept of finite automata to record the transitions of objects' spatial relations. From the final states of the finite automata, the higher-level semantics of spatial relation changes between the objects in a video can be inferred.

The rest of the paper is organized as follows. We first review the 3D C-string approach in Section 2. In Section 3, the video semantic model for fixed-size and varying-size objects is introduced. Then, the spatio-temporal relation inference algorithm is described. Finally, conclusions are given in Section 4.

2. 3D C-STRING APPROACH

The knowledge structure of 3D C-string [4] uses the projections of the objects to represent the spatial and temporal relations among the objects in a video. The objects in a video are projected onto the x-, y-, and time-axes to form three strings representing the relations and relative positions of the projections in the x-, y- and time-axes, respectively. The projections of an object onto the x-, y- and time-axes are called x-, y-,

IEEE computer society

The 3rd Intetnational Conference on Innovative Computing Information and Control (ICICIC'08)
978-0-7695-3161-8/08 $25.00 © 2008 IEEE
Authorized licensed use limited to: MIT Libraries. Downloaded on September 28, 2009 at 22:08 from IEEE Xplore. Restrictions apply.

and time-projections, respectively. In the knowledge structure of 3D C-string, there are 13 relations for one-dimensional intervals for each dimension. For the x, y and time dimensions, each of them has 13 relations as shown in Table 1.where B(P) and E(P) are the begin-bound (beginning point) and end-bound (ending point) of the x-, y- or time-projection of object P. For example, in the x and y dimensions, P<Q represents that the projection of object P is before that of object Q. In the time dimension, P<Q denotes that object P disappears before object Q appears. Each object is approximated by a minimum bounding rectangle (MBR) whose sides is parallel to the x- and y-axes and keeps track its initial location and size, and record the information about the motions and size changes of the objects in 3D C-string.

Table 1. The definition of 13 spatial operators.

Notations	Conditions	Notations	Symmetric conditions
$P < Q$	$E(P)<B(Q)$	$P <^* Q$	$E(Q)<B(P)$
$P \mid Q$	$E(P)=B(Q)$	$P \mid^* Q$	$E(Q)=B(P)$
P / Q	$B(P)<B(Q)<E(P)<E(Q)$	$P /^* Q$	$B(Q)<B(P)<E(Q)<E(P)$
$P [Q$	$B(P)=B(Q), E(P)>E(Q)$	$P [^* Q$	$B(Q)=B(P), E(Q)>E(P)$
$P = Q$	$B(P)=B(Q), E(P)=E(Q)$	$P =^* Q$	Same as left
$P \% Q$	$B(P)<B(Q), E(P)>E(Q)$	$P \%^* Q$	$B(Q)<B(P), E(Q)>E(P)$
$P] Q$	$B(P)<B(Q), E(P)=E(Q)$	$P]^* Q$	$B(Q)<B(P), E(Q)=E(P)$

3. VIDEO SEMANTIC MODEL

In this section, we propose a video semantic model to infer the relation changes between the objects in a video represented by a 3D C-string. First, we reconstruct the 3D C-string to an object list [4]. In the list, every object has the information including its initial location and size, motion vectors and ratios of size changes. Secondly, we generate the one-dimension relations for each object pair and use an integer value to present each one-dimensional relation as listed in Table 2. For any two objects. There are 13 possible one-dimensional relations between their x- or y- projections. Therefore, there are 169 two-dimensional relations between two objects. Thirdly, we divide 169 two-dimensional relations into six spatial relations, namely, *disjoint*, *join*, *overlap*, *contain*, *belong* and *equal*.

Table 2. The integer values of 1D relations.

Relation	P<Q	P|Q	P/Q	P]Q	P%Q	P[Q	P=Q
Value	1	2	3	4	5	6	7
Relation	Q[P	Q%P	Q]P	Q/P	Q|P	Q<P	Q=P
Value	8	9	10	11	12	13	7

Finally, by inferring the spatial relations for each pair of objects, we can easily obtain the higher-level semantics of their spatial relation changes. The *spatio-temporal relation inference algorithm* is described in detail as follows.

Algorithm: spatio-temporal relation inference

Input: a 3D C-string of a video

Output: the higher-level semantics of spatial relation changes between any two objects

1. Apply *3D C-string video reconstruction algorithm* [4] to the given u-, v- and t-string. The algorithm returns the initial locations, sizes, and the motion lists of all objects in the u-string (or v-string). It also returns the starting frame numbers, the length of duration, and change lists of all objects in the t-string.
2. For each pair of objects P and Q, perform steps 3~7.
3. Set the spatio-temporal list of objects P and Q to null (empty).
4. Determine the temporal relation between objects P and Q by their starting frame numbers and length of duration.
5. If the temporal relation between objects P and Q is equal to one of the following relations: "P/Q", "P]Q", "P%Q", "Q[P", "P=Q", "P[Q", "Q%P", "Q]P", or "Q/P", record the starting and ending frame numbers of their concurrent (overlapped) period. Call the *relation inference for concurrent objects algorithm* with objects P, Q and their concurrent period as input parameters, and append the returned spatio-temporal list and associated duration to their spatial-temporal list.
6. Generate a sequence of spatial relations between objects P and Q as shown in Table 2.
7. Input the sequence of spatial relations to the corresponding finite automaton (described later). Output the meanings of the final state for objects P and Q.

In the step 5, we consider how to compute the spatial relations for two concurrent objects. If the sizes of objects P and Q are not changed during the motions, there are three possible cases when two concurrent objects P and Q change their spatial relations in the one-dimensional space. They are listed as follows: (1) The size of object P is bigger than that of object Q; (2) The size of object P is equal to that of object Q; (3) The size of object P is smaller than that of object Q.

Let's consider how the spatial relation between objects P and Q is changed for the case (1) where the size of object P is bigger than that of object Q. For example, if the spatial relation between objects P and Q is P<Q, initially. Then object P is moving toward the positive direction of the x-axis and object Q is moving toward the negative direction of the x-axis. The spatial relation between objects P and Q is changed gradually from P<Q, P|Q, P/Q, P]Q, P%Q, P[Q, Q/P, Q|P, to Q<P. So, we can define the ranks of the spatial relations by the order of changes. That is, rank(P<Q) is 1, rank(P|Q) is 2, and so on. If the sizes of objects P and Q are not changed during the motions, the ranks of

IEEE computer society

The 3rd Intetnational Conference on Innovative Computing Information and Control (ICICIC'08)
978-0-7695-3161-8/08 $25.00 © 2008 IEEE
Authorized licensed use limited to: MIT Libraries. Downloaded on September 28, 2009 at 22:08 from IEEE Xplore. Restrictions apply.

their spatial relations should be changed one by one. Similarly, we can define the ranks of the spatial relations between objects P and Q for the cases (2) and (3) as shown in Table 3.

Table 3. The ranks of the spatial relations between objects P and Q.

Case	Relation	P<Q	P|Q	P/Q	P]Q	P%Q	P[Q	Q/P	Q|P	Q<P
(1)	Rank	1	2	3	4	5	6	7	8	9
Case	Relation	P<Q	P|Q	P/Q	P=Q			Q/P	Q|P	Q<P
(2)	Rank	1	2	3	5			7	8	9
Case	Relation	P<Q	P|Q	P/Q	Q[P	Q%P	Q]P	Q/P	Q|P	Q<P
(3)	Rank	1	2	3	4	5	6	7	8	9

For the concurrent objects, if objects neither P nor Q change its size, and the rank difference before and after a motion are greater than one, we need to compute all the relations between objects P and Q during the motion. When either object P or Q changes its size during a motion, we need to compute all the relations between objects P and Q during the motion if (1) the rank difference is greater than one or (2) the rank difference is equal to one and there exist rank 4, 5 or 6 either before or after the motion. The *relation inference for concurrent objects algorithm* is described in detail as follows.

Algorithm: relation inference for concurrent objects

Input:Objects O_i, O_j, T_s (the starting frame number of the concurrent period), and T_e (the ending frame number of the concurrent period)

Output: the spatio-temporal relation list of objects O_i and O_j in the concurrent period. (There are two fields for each element in the list: one records a relation, and the other records the length of the duration associated with the relation.)

1. If the starting frame number of objects O_i or O_j is smaller than T_s, compute the location and size of that object in the frame T_s.
2. Determine the spatial relation SR between both objects in the x (or y) dimension by inputting the locations and sizes of objects O_i and O_j in frame T_s.
3. Insert a new element with its relation field equal to SR and the duration field equal to 0 into the spatio-temporal relation list of objects O_i and O_j.
4. Retrieve the time points (measured by the frame) in which objects O_i changes its states in the concurrent period, and also retrieve the time points in which objects O_j changes its states in the concurrent period.
5. If the values of any two time points are equal, merge them into one. If T_e is not included, have it included. Sort the time points collected in steps 4 in increasing order.
6. For each time point in step 5, if (1) SR is the relation immediately next to SR' in Table 3 or (2) |rank(SR)-rank(SR')|=1 and both ranks are in the range of 1~3 or 7~9, compute the duration of the SR and SR'. Add the length of the duration of SR to the duration field of the last element in the list and append a new element with its relation field equal to SR' and the duration field equal to the length of the duration of SR' to the list.
7. Otherwise, compute all the spatial relations between objects O_i and O_j during the period and append those relations with their associated duration to the list.

After finishing the relation inference for concurrent objects algorithm, we can get two spatio-temporal relation changing lists for each pair of objects P and Q in the x- and y-dimension. Based on the spatial relation sequence of objects P and Q, we can use a finite automaton to infer the spatial relation changes in the video and thus get the higher-level semantics of spatial relation changes between the objects. Objects P and Q may change their sizes in the video. Let's first consider that both objects do not change their sizes (fixed-size) in the video. There are four possible cases when objects P and Q change their spatial relations in the two-dimensional space. They are listed in Table 4, where $S_{P,X}$ denotes the size of the x-projection of object P and $S_{P,Y}$ denotes the size of its y-projection.

Table 4. Possible spatial relations.

Case	Constraints of the sizes of x-(y-) projections	Possible spatial relations between objects P and Q
(I)	($S_{P,X}>S_{Q,X}$ and $S_{P,Y}>S_{Q,Y}$) or ($S_{P,X}>S_{Q,X}$ and $S_{P,Y}=S_{Q,Y}$) or ($S_{P,X}=S_{Q,X}$ and $S_{P,Y}>S_{Q,Y}$)	*disjoin*, *join*, *overlap*, and *contain*
(II)	($S_{P,X}<S_{Q,X}$ and $S_{P,Y}<S_{Q,Y}$) or ($S_{P,X}<S_{Q,X}$ and $S_{P,Y}=S_{Q,Y}$) or ($S_{P,X}=S_{Q,X}$ and $S_{P,Y}<S_{Q,Y}$)	*disjoin*, *join*, *overlap*, and *belong*
(III)	($S_{P,X}>S_{Q,X}$ and $S_{P,Y}<S_{Q,Y}$) or ($S_{P,X}<S_{Q,X}$ and $S_{P,Y}>S_{Q,Y}$)	*disjoin*, *join*, and *overlap*
(IV)	$S_{P,X}=S_{Q,X}$ and $S_{P,Y}=S_{Q,Y}$	*disjoin*, *join*, *overlap* and *equal*

For the case (I), there exist only four possible spatial relations between objects P and Q, namely, *disjoin*, *join*, *overlap*, and *contain*. That is, it is impossible for both objects to have the spatial relations of *belong* and *contain*. Since there are four possible spatial relations in case (I), we can construct the corresponding finite automaton with four initial states. Similarly, the number of initial states for cases (II), (III), (IV) are 4, 4, and 3, respectively. The finite automata for cases (II), (III), and (IV) are similar to that for case (I), we only present the finite automaton of disjoin for case (I) as shown in Fig. 1.

Fig. 1 shows the finite automaton starting from the initial state of *disjoin* for case (I), where each state is a final state. The finite automaton changes its state according to the inputs, that is, the spatial relation sequence of objects P and Q. For example, initially, if the relation between P and Q is *disjoin*, the finite automaton stays in D_1. Afterwards, if the spatial

IEEE computer society

The 3rd Intetnational Conference on Innovative Computing Information and Control (ICICIC'08)
978-0-7695-3161-8/08 $25.00 © 2008 IEEE
Authorized licensed use limited to: MIT Libraries. Downloaded on September 28, 2009 at 22:08 from IEEE Xplore. Restrictions apply.

relation between P and Q is changed to *join*, the finite automaton changes its state to D_2. Then, if the spatial relation between P and Q is changed to *overlap*, the finite automaton changes its state to D_4.

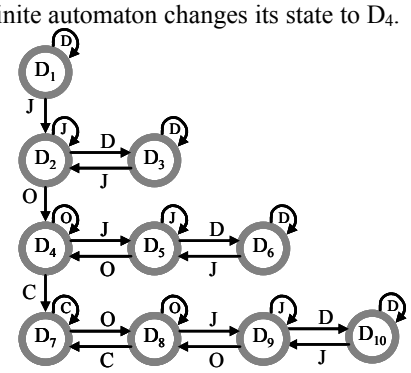

Fig. 1. The finite automaton starting from disjoin for case (I).

Similarly, the finite automaton changes its state according to the spatial relation sequence between P and Q. For example, if we have a spatial relation sequence, DJD, the finite automaton starts from D_1, passes through D_2, and stops at D_3. Therefore, the final state for the spatial relation sequence, DJD, is D_3. Similarly, if we have a spatial relation sequence, DJOCOJD, the finite automaton starts from D_1, sequentially passes through D_2, D_4, D_7, D_8, D_9, and stops at D_{10}. So, the final state for the spatial relation sequence, DJOCOJD, is D_{10}.

Table 5. The number of states for each finite automaton

Sizes of objects	Case	Finite automaton starting from the following spatial relation						Total number of states
		Disjoin	*Join*	*Over-lap*	*Contain*	*Belong*	*Equal*	
Fixed-Size	(I)	10	26	26	10	NA	NA	72
	(II)	10	26	26	NA	10	NA	72
	(III)	6	11	6	NA	NA	NA	23
	(IV)	10	26	26	NA	NA	10	72
Varying-size		200	1066	2364	675	675	958	5938

Now, let's consider the varying-size case in which the objects in a video may change their sizes. Since there are six possible spatial relations, we have to construct six finite automata, each of which starts from a spatial relation. The major difference between the finite automaton for the varying-size objects and that for the fixed-size objects is that the states transitioning from a state is richer since the size of an object can be changed. For example, *overlap* can be changed to *join*, *contain*, *belong*, or *equal*, *contain* can be changed to *overlap* or *equal*, *belong* can be changed to *overlap* or *equal*, *equal* can be changed to *overlap*, *contain* or *belong*.

For simplicity, we only summary the number of final states for each finite automaton is listed in Table 5, where NA stands for "not available".

4. CONCLUDING REMARKS

In this paper, we propose a video semantic model to get the higher-level semantics of spatial relation changes between the objects in a video represented by a 3D C-string. We use the concept of finite automata to record the transitions of objects' spatial relations. From the final states of the finite automaton, the higher-level semantics of spatial relation changes between the objects in a video can be inferred. Users may use higher-level semantics to retrieve a video from a video database management system. Therefore, our new semantic model can be easily applied to an intelligent video database system, and to reason about spatial relation changes between the objects in a video.

REFERENCES

[1] L. Bai, S. Lao, G.J.F. Jones and A.F. Smeaton, "A Semantic Content Analysis Model for Sports Video Based on Perception Concepts and Finite State Machines," *Multimedia and Expo, 2007 IEEE International Conference*, July 2007, pp.1407-1410.

[2] D. Djordjevic and E. Izquierdo, "An Object- and User-Driven System for Semantic-Based Image Annotation and Retrievalm," *IEEE Trans. On Circuits and Systems for Video Technology*, Vol. 17, Issue 3, 2007, pp.313-323

[3] W. Hu, D. Xie, Z. Fu, W. Zeng and S. Maybank, "Semantic-Based Surveillance Video Retrieval", *IEEE Trans. Image Processing*, Vol.16, Issue 4, 2007, pp.1168-1181.

[4] Anthony J.T. Lee, H.-P. Chiu and P. Yu, "3D C-string: A New Spatio-temporal Knowledge Structure for Video Database Systems," *Pattern Recognition*, Vol.35, No. 11, 2002, pp. 2521-2537.

[5] M. Worring and G. Schreiber, "Semantic Image and Video Indexing in Broad Domains", *IEEE Trans. on Multimedia*, Vol. 9, Issue 5, 2007, pp.909-911.

IEEE computer society

978-0-7695-3161-8/08 $25.00 © 2008 IEEE

Authorized licensed use limited to: MIT Libraries. Downloaded on September 28, 2009 at 22:08 from IEEE Xplore. Restrictions apply.