Similarity Retrieval of Video Database Based on 3D Z-string

Ping Yu, Chein-Shung Hwang, NaiWen Kuo
Dept. of Information Management, Chinese Culture University
Taipei, Taiwan
{yp, cshwang, neven}@faculty.pccu.edu.tw

***Abstract*—Recently, the video retrieval processing is concerned with retrieving videos that are relevant to the users' requests from a large collection of videos, referred to as a video database. We have proposed 3D Z-string to represent symbolic videos accompanying with the string generation and video reconstruction algorithms. In this paper, we proposed the spatial-temporal similarity retrieval approach of vides in 3D Z-string. Our approach defines a set of user assigned weights, based on the factors of spatial-temporal relations of object pairs in a video, in order to rank the retrieval videos. We use dynamic programming to calculate the similarity measures and propose the similarity retrieval algorithm. By providing various criterion of similarity between videos to match user requirement, our proposed similarity retrieval algorithm has discrimination power about different criteria.**

Keywords-Video database; Spatial-temporal inference; Similarity retrieval; 3D Z-string

I. INTRODUCTION *(HEADING 1)*

With the advances in information technology, videos have been promoted as a valuable information resource. Because of its expressive power, videos are an appropriate medium to show dynamic and compound concepts. Recently, there has been widespread interest in various kinds of database management systems for managing information from videos. The video retrieval problem is concerned with retrieving videos that are relevant to the users' requests from a large collection of videos, referred to as a video database.

Over the last decade, many image/video indexing and retrieval methods have been proposed, for example, OVID, KMED, QBIC, VideoQ, VideoText, and VideoQA etc. [1], The T2D-histogram [2] and Pixel Change Ratio Map (PCRM) [3] used the histogram method to index the videos. Su [4] used the motion vectors embedded in MPEG bit streams and do not consider the shape of a moving object and its corresponding trajectory. Hsieh [5] also proposed a hybrid motion-based video retrieval system to retrieve desired videos from video databases through trajectory matching. Spatial-temporal visual map (STVM) [6] defined the spatial-temporal visual similarity to rerank the text-retrieval results and find new results. Snoek [7] proposed an automatic video retrieval method based on high-level concept detectors that provided three strategies, namely: text matching, ontology querying, and semantic visual querying to select a relevant detector from the video database. To retrieve desired videos from a video database, one of the most important methods for discriminating videos is the perception of the objects and the spatial-temporal relations that exist between the objects in a video. To represent the spatial relations between the objects in a symbolic image, many iconic indexing approaches and image retrieval algorithms have been proposed. Such as, 2D string, 2D G-string, 2D C-string, 2D C^+-string, unique-ID-based matrix, GPN matrix, virtual image , BP matrix, and 2D Z-string [8]. To represent the spatial and temporal relations between the objects in a symbolic video, many iconic indexing approaches, extended from the notion of 2D string to represent the spatial and temporal relations between the objects in a video, have been proposed. For example, 2D B-string, 2D C-Tree, 9DLT strings, 3D-list, 3D C-string, and 3D Z-string [9]. The 3D Z-string, extended from the 2D Z-string, used the projections of objects to represent spatial and temporal relations between the objects in a video. Since there are no cuttings between objects in the 3D Z-string, compare to the 3D C-string, the integrity of objects is preserved. The 3D Z-string is more compact and efficient in terms of storage space and execution time than 3D C-string.

The capability of similarity retrieval is important in video database management systems. In 3D C-string similarity retrieval [10], used the time interval sets of similar spatial relation sequence and temporal relations for each pair of objects in a video and defined various types of similarity measures (type-std, Spatial, Temporal, and Duration) to construct the association graph, which connected by an edge which associates with an interval set. The similarity is the intersection of the interval sets to find the largest type-std clique. Therefore, the exactly match would lose some similar information, and the similarity retrieval cost of 3D C-string is NP-hard. Accordingly, in this paper, we proposed a new similarity approach that consists two phases. First, we defined the weight set of spatial-temporal similarity of the pairs of objects between the videos that can let users assign the requirement of spatial-temporal relations similarity. The spatial-temporal similarity separated to two cross-level that avoid loosing of similarity information from exactly matched approach. Second, we use the dynamic programming approach to calculate the total similarity between videos, and propose the similarity retrieval algorithm. By providing various weights of similarity between pairs of objects in a video to match user query requirement, this similarity retrieval algorithm has discrimination power about different criteria.

978-0-7695-3643-9/09 $25.00 © 2009 IEEE
DOI 10.1109/ISECS.2009.190

The remainder of this paper is organized as follows. We first review the 3D Z-string approach of representing symbolic videos in Section 2. In Section 3, we describe the spatial-temporal relation inference algorithms from which all relation sequences between objects can be easily derived. Then we discuss the similarity retrieval algorithm based on user assign weighted similarity between videos in Section 4. In Section 5, the results of performance experiments are presented. Finally, concluding remarks are made in Section 6.

II. 3D Z-STRING APPROACH

In the knowledge structure of 3D Z-string [9], we use the projections of objects to represent the spatial-temporal relations between 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. These three strings are called u-, v- and t-strings. The projections of an object onto the x-, y- and time-axes are called x-, y-, and time-projections, respectively. There are 13 relations between two x- (y- or time-) projects in the 3D Z-string as sowing in Table 1, where B_P and E_P are the begin-bound and end-bound of the x- (y- or time-) projection of object P, respectively. For example, in the x (or y) dimension, $P<Q$ 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.

TABLE I. THE DEFINITION OF 13 SPATIAL-TEMPORAL 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
$P \% Q$	$B(P)<B(Q)$, $E(P)>E(Q)$	$P \%^* Q$	$B(Q)<B(P)$, $E(Q)>E(P)$
$P] Q$	$B(P)<B(Q)$, $E(P)=E(Q)$	$P]^* Q$	$B(Q)<B(P)$, $E(Q)=E(P)$

III. THE PRESENTATIONS OF THE SPATIAL-TEMPORAL RELATIONS

First of all, we can use the definition of spatial-temporal relation in the Table 1 to determine the spatial relation between the x- (or y-) projections of a pair of objects. Similarly, we also can determine the temporal relation between the time-projections of a pair of objects. Second, we define the spatial relation sequence SRS to precisely record the spatial relation changes and temporal relation TR_{PQ} to record the temporal relation between object pairs in a video as following.

Definition 1: If P and Q are the object pair in video V, a spatial relation sequence $SRS^u = SR^u_1, SR^u_2, \ldots, SR^u_n$ (or $SRS^v = SR^v_1, SR^v_2, \ldots, SR^v_n$) where SR^u_k (or SR^v_k) means that the spatial relation between P and Q in the x (or y) dimension. We said the SRS^u_{PQ} (or SRS^v_{PQ}) is the spatial relation sequence between P and Q in the x (or y) dimension of video V.

Definition 2: If P and Q are the object pair in video V, a temporal relation TR_{PQ} means the temporal relation between the time-projection of object P and that of object Q in video V.

For example, there are two objects A and B, if the spatial relation between objects A and B is "A%B" in frames 1 and 2, and is "A]B" in frame 3. The spatial relation sequence of objects A and B is SRS_uAB ={"A%B", "A]B"}, and the temporal relation TR_{AB} = {"A=B"}. Therefore, we can obtain all the spatial relation sequences for each pair of objects.

IV. SIMILARITY RETRIEVAL

In this section, we first define the similarity between videos based on the spatial-temporal relations between objects in the videos, which allow a user to assign different levels of weights to the spatial and temporal relations and to calculate the similarity between videos. Then, we propose the similarity retrieval algorithm, which uses the dynamic programming approach to calculate the similarity between videos.

Assume that a pair of objects (P, Q) in a video V' matches a pair of objects (P, Q) in another video V. We use the following notations to define the spatial-temporal relations.

Definition 3: Given two spatial relation sequences $SRS = SR_1, SR_2, \ldots, SR_n$ and $SRS' = SR'_1, SR'_2, \ldots, SR'_m$ where $n \geq m > 0$, if $SR_{j_i} = SR'_i, j_1 < j_2 < \ldots < j_m$, for all i=1, 2, …, m, we can say that SRS' is a sub-sequence of SRS. The (P, Q) is called a *spatially similar pair* between videos V' and V, if SRS'^u_{PQ} and SRS'^v_{PQ} both are the sub-sequences of SRS^u_{PQ} and SRS^v_{PQ}, respectively.

We also use the category sequence of the spatial relations to present the topology relations between P and Q. The concept of the categories of spatial relations was proposed by the 2D C-string. They divided 169 spatial relations, from 13*13 relations in the x and y dimension, into five spatial categories, namely, *disjoin*, *join*, *overlap*, *contain*, and *belong*.

Definition 4: If P and Q are the object pair in video V, a spatial category sequence is a sequence of $SC_1, SC_2, \ldots, SC_n$, where SC_i is the category of the ith spatial relation between P and Q in the x (or y) dimension. SCS_{PQ} is the spatial category sequence between P and Q in the x (or y) dimension of video V.

Definition 5: Given two spatial category sequences $SCS = SC_1, SC_2, \ldots, SC_n$ and $SCS' = SC'_1, SC'_2, \ldots, SC'_m$ where $n \geq m > 0$, if $SC_{ji} = SC'_i, j_1 < j_2 < \ldots < j_m$, for all i=1,2, …,m, we can say that SCS' is a sub-sequence of SCS. (P, Q) is called a *spatially c-similar pair* between videos V' and V, if SCS'_{PQ} is the sub-sequences of SCS_{PQ}.

Definition 6: Lets temporal intervals T_P and T'_P denote the size of the time-projection of object P in video V and V', respectively. (P, Q) is called a *temporally i-similar pair* between videos V' and V, if $T'_P = T_P$ and $T'_Q = T_Q$.

Definition 7: Let TR'_{PQ} and TR_{PQ} be the temporal relations of the time-projection of object P and that of object Q in video

The work of Ping Yu was supported in part by the National Science Council, Taiwan, under grant number NSC-97-2218-E-034-003

V' and V, (P, Q) is called a *temporally similar pair* between videos V' and V, if $TR'_{PQ} = TR_{PQ}$.

By defining the spatial-temporal similarity between an object pair, we can define different criteria such as spatial-temporal relation or category to measure the similarity degree between the object pair. The similarity between (P, Q) in video V' and (P, Q) in video V can be the combinations of different levels of those criteria. (P, Q) is called a *similar pair*, and objects P and Q are called *matched objects*.

Since video data contain very rich spatial-temporal information, users may extract different spatial-temporal levels of information according to their interests. Therefore, we can define the similarity between an object pair (P, Q) as in (1), where the sum of W_{SRS}, W_{SCS} ,W_T, and W_{TR} is equal to one.

$$Similarity(P, Q) = W_{SRS} * Sim^{S1}_{(P, Q)} + W_{SCS} * Sim^{S2}_{(P, Q)} + W_T * Sim^{T1}_{(P, Q)} + W_{TR} * Sim^{T2}_{(P, Q)} \quad (1)$$

To find the similarity between videos V' and V, we must consider all possible matched object sets from both videos. However, there are a large number of matched object sets, and it seems difficult to find all of them. We solve such a problem by the dynamic programming approach.

Let $O = O_1, O_2, \ldots, O_n$, be objects contained in query video V, where n is the number of objects of V, and $O = O'_1, O'_2, \ldots, O'_n$, be matched objects in video V' that O_1, where (O_i, O'_i) is an similar object pair, i=1, 2, …, n. We can form a weighted directed graph $G = (V, E)$, where G contains n vertices v_1, $v_2,\ldots,v_n$, v_i denotes object pair (O_i, O'_i), a weight function w: $E \rightarrow R$ maps an edge to the real-valued similarity. We wish to find, for each path of vertices v_i to v_n, where i=1 to n-1, to calculate a maximum similarity path, where the similarity of a path is the sum of the similarities between each vertex. The weighted directed graph $G = (V, E)$ can be represented by an adjacency-matrix W, where W is an $n*n$ matrix, and w_{ij} represents W the similarity between objects O_i and O_j if objects O_i and O_j is a similar pair; otherwise, w_{ij} is equal to 0.

The tabular output of the maximum similarity algorithm presented is an $n* n$ matrix D= (d_{ij}), where entry d_{ij} contains maximum similarity of a path from vertex i to vertex j. To solve the maximum similarity problem on an input adjacency matrix, we need to compute not only the path of maximum similarities but also a predecessor matrix R= (r_{ij}), where r_{ij} is NIL, if $i \geq j$; otherwise, r_{ij} represents a predecessor of j on a path starting from i. For each vertex $i \in V$, we define the predecessor subgraph of G for i as $G_{r,i} = (V_{r,i}, E_{r,i})$, where $V_{r,i} = \{ j \in V: r_{ij} \neq \text{NIL}$ and $i < j \} \cup \{j\}$, and $E_{r,i} = \{(r_{ij}, j): V_{r,i}$ and $r_{ij} \neq \text{NIL}$ and $i < j \}$.

We denote the matrices by uppercase letters, such as R or D, and their individual elements by subscripted lowercase letters, such as r_{ij} or d_{ij}. Some matrices will have parenthesized superscripts, as in $R^{(m)}=(r_{ij}^{(m)})$ or $D^{(m)}=(d_{ij}^{(m)})$, where m is the number of iteration.

Let $d_{ij}^{(m)}$ be the maximum similarity path of from vertex j to vertex $j+m$ that contains at most m-1 edges in the iteration of m. When m= 0, there is a path from i to i. Thus, $d_{ij}^{(0)}=0$, and $d_{ij}^{(1)}$ is the similarity between object i and j as the weight w_{ij}. For $m \geq 1$, we compute $d_{ij}^{(m)}$ as the maximum of $d_{ij}^{(m-1)}$, the maximum similarity from j to $j+m$-1 consisting of at most m-1 edges, where $j+ m< n$, obtained by looking at predecessor m-1 of j. Thus, we recursively define $d_{ij}^{(m)} = d_{ij}^{(m-1)} + d_{i(j+1)}^{(m-1)} + w_{j(j+m-1)} - d_{i(j+1)}^{(m-2)}$, when $r_{ij} \neq$NIL, and $d_{ij}^{(m)} = d_{ij}^{(m-1)} + d_{i(j+1)}^{(m-1)} - d_{i(j+1)}^{(m-2)}$, when r_{ij}=NIL.

By taking as the input matrix $D^{(1)}=W$, we now compute a series of matrices $D^{(1)}$, $D^{(2)}$, …, $D^{(n-1)}$, m=1, 2, …, n-1. The main idea of the algorithm is to extend the maximum similarity path computed so far one more edge for each iteration. The matrix $D^{(n-1)}$ contains the maximum similarity of each path in the $d_{i(i+1)}^{(n-1)}$, where i=1, 2, …, n-1, that can return a similarity rank list. The *video retrieval algorithm* is described in detail in Fig. 1. The *maximum similarity algorithm* is described in detail in Fig. 2.

(1)

Algorithm: similarity retrieval
Input: the assigned weights of W_{SRS}, W_{SCS}, W_T, and W_{TR} of spatial-temporal similarity, and two videos V' and V
Output: the similarity rank list between V' and V.

1. Computing the temporal relation and spatial sequence for each object pair in video V'.
2. Construct the n-vertex weighted directed graph $G = (V, E)$ for videos V, where n is the number of objects in V, and the weight in an edge is the similarity of the matched object pair between V' and V.
3. Construct the similarity martix $D^{(1)}$, where $d_{ij}^{(1)}= w_{ij}$ for all vertices $i, j \in V$.
4. Call the *maximum similarity algorithm* with $D^{(1)}$.
5. Remove the object list from similarity rank list that be contained in the others.

Figure 1. Similarity retrieval algorithm.

Algorithm: maximum similarity
Input: the similarity matrix $D^{(1)}$
Output: similarity rank list

1. $i \leftarrow 0, j \leftarrow 0, k \leftarrow 0$
2. **for** $m \leftarrow 2$ **to** n-1 **do** // n is the number of objects
3. **for** $i \leftarrow 1$ **to** n-m **do** // i is the index of row
4. **for** $j \leftarrow i+1$ **to** n-m+1 **do** // j is the index of column
5. $r^m(i,j) \leftarrow r^{m-1}(i,j) \cup r^{m-1}(i,j+1)$
6. **if** $d_{i(j+1)}^{(1)} \neq 0$ **then** // $r_{j(j+1)} \neq$NIL
7. $d_{ij}^{(m)} \leftarrow d_{ij}^{(m-1)} + d_{i(j+1)}^{(m-1)} - d_{i(j+1)}^{(m-2)} + d_{j(j+m-1)}^{(1)}$
8. **else** // $r_{j(j+1)}$=NIL
9. $d_{ij}^{(m)} \leftarrow d_{ij}^{(m-1)} + d_{i(j+1)}^{(m-1)} - d_{i(j+1)}^{(m-2)}$
10. **end if**
11. **end for**
12. **end for**
13. **end for**

Figure 2. Maximum similarity algorithm.

V. PERFORMANCE ANALYSIS

To show the effectiveness of our proposed approach with that of the 3D C-string and *9DLT* approach, we show our *similarity retrieval algorithm* on the real videos to present the precision versus recall analysis. In the example video database, there are 100 videos of soccer games. All videos are one minute long. Typically, a video of one minute long contains 1800 frames. To represent the movements of objects, at least a frame should be indexed for every 10 frames. To compare the retrieval capability of different weights of similarity, we assign three sets of similarity weights: 3DZ-w1) W_{SRS} =0, W_{SCS} =0.5, W_T =0, and W_{TR}= 0.5; 3DZ-w2) W_{SRS} =0.3, W_{SCS} =0.2, W_T =0.3, and W_{TR}= 0.2; and 3DZ-w3) W_{SRS} =0.8, W_{SCS} =0, W_T =0.1, and W_{TR}= 0.1.

Fig. 2 illustrates the precision versus recall for the 3D Z-string, 3D C-string and 9DLT approaches. The result of the 3DZ-w3 approach is better than that of the 3D C-string type-322 approach, because the 3DZ-w3 approach can provide a more flexible way to compute the similarity in spatial-temporal relation. The 3D C-string type-322 approach which is better than that of the 3DZ-w2 and 3DZ-w1 approaches, because both approaches release some spatial relation constraints to spatial category. The results of type-300 query and 9DLT-string are quite closely, because both approaches only use the spatial relationships to query the database. In summary, both 3D Z-string and 3D C-string approaches can provide various types of similarity between videos and have discrimination power about different criteria. However, the 3D Z-string approach can provide a more flexible way to retrieve similar videos from the database and to meet user requirement.

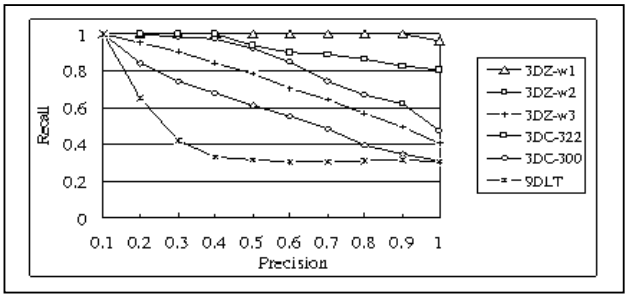

Figure 3. Fig. 4. Precision vs. recall.

VI. CONCLUDING REMARKS

In this paper, we proposed a new video retrieval method based on the 3D Z-string to avoid the shortcomings of the video retrieval method of the 3D C-string. Our proposed approach consists of two phases. First, we infer the spatial relation sequence and temporal relations for each object pair in a video, and use the inferred temporal and spatial relation sequences associated with the weights of similarity to calculate the similarity between objects. Second, we use the dynamic programming approach to compute the similarity between videos and find a list of ranked similar videos. We also show that different objects with same attributes can group as a similarity list to calculate the similarity in the different number of objects between videos. By providing different weights of similarity, our proposed similarity retrieval algorithm has discrimination power about different criteria. Our proposed approach can be easily applied to an intelligent video database management system to infer spatial and temporal relations between the objects in a video and to retrieve the videos similar to a query video from a video database.

REFERENCES

[1] W. Al-Khatib, Y. F. Day, P. B. Berra, "Semantic Modeling and Knowledge Representation in Multimedia Databases," IEEE Trans. on Knowledge and Data Engineering, Vol. 11, No. 1, 1999, pp.64-80.

[2] D.-Y. Chena, S.-Y. Leea, H.-Y. Mark Liao, "Robust video sequence retrieval using a novel object-based T2D-histogram descriptor," J. Vis. Commun. Image R., vol. 16, 2005, pp. 212-232

[3] H. Yi, D. Rajan, L-T. Chia, "A new motion histogram to index motion content in video segments," *Pattern Recognition Letters,* vol. 26, 2005, pp. 1221-1231.

[4] C.-W. Su, H.-Y.M. Liao, H.-R. Tyan, C.-W. Lin, D.-Y. Chen, and K.-C. Fan, "Motion flow-based video retrieval," IEEE Trans. on Multimedia, vol. 9, no. 6, Oct. 2007, pp. 1193-1201

[5] J.-W. Hsieh, S.-L. Yu, and Y.-S. Chen, "Motion-based video retrieval by trajectory matching," IEEE Trans. on Circuits and Systems for Video Technology, vol 16, no. 3, Mar. 2006, pp. 396- 409.

[6]]H.-B. Luan, S.-X. Lin, S. Tang, S.-Y. Neo, and T.-S. Chua, "Interactive spatio-temporal visual map model for web video retrieval," Proc. of IEEE Intl. Conf. on Multimedia and Expo, 2-5 July 2007, pp. 560-563

[7] C.G.M. Snoek, B. Huurnink, L. Hollink,M. de Rijke, G.. Schreiber, and M.Worring, "Adding semantics to detectors for video retrieval," IEEE Trans. on Multimedia, vol. 9, no.5, Aug. 2007, pp.975-986.

[8] A. J. T. Lee, H. P. Chiu, "2D Z-string: a new spatial knowledge representation for image databases," Pattern Recognition Letter, vol. 24, 2003 , pp. 3015-3026.

[9] A. J. T. Lee, P. Yu, H.P. Chiu, "3D Z-string: a new knowledge structure to represent spatial-temporal relations between objects in a video," Pattern Recognition Letters, vol. 26, pp.2500-2508, 2005.

[10] A. J. T. Lee, P. Yu, H.P. Chiu, "Similarity Retrieval of Videos by Using 3D C-String Knowledge Representation," Journal of Visual Communication and Image Representation, vol. 16, pp.749-773, 2005.