Temporal Segmentation of Video Objects for Hierarchical Object-Based Motion Description

Yue Fu, Ahmet Ekin, *Student Member, IEEE*, A. Murat Tekalp, *Senior Member, IEEE*, and Rajiv Mehrotra

***Abstract*—This paper describes a hierarchical approach for object-based motion description of video in terms of object motions and object-to-object interactions. We present a temporal hierarchy for object motion description, which consists of low-level elementary motion units (EMU) and high-level action units (AU). Likewise, object-to-object interactions are decomposed into a hierarchy of low-level elementary reaction units (ERU) and high-level interaction units (IU). We then propose an algorithm for temporal segmentation of video objects into EMUs, whose dominant motion can be described by a single representative parametric model. The algorithm also computes a representative (dominant) affine model for each EMU. We also provide algorithms for identification of ERUs and for classification of the type of ERUs. Experimental results demonstrate that segmenting the life-span of video objects into EMUs and ERUs facilitates the generation of high-level visual summaries for fast browsing and navigation. At present, the formation of high-level action and interaction units is done interactively. We also provide a set of query-by-example results for low-level EMU retrieval from a database based on similarity of the representative dominant affine models.**

***Index Terms*—Motion description, parametric motion model, video browsing and navigation, video indexing, video summary.**

I. Introduction

THE rapid proliferation of multimedia applications presents a growing need for new effective representations of video that allow not only compact storage but also content-based functionalities such as object-based navigation, search and browsing. The size and rich content of video data makes the organization, indexing and management of visual databases for efficient and effective browsing and retrieval a challenging task. Most of the existing systems for video browsing and retrieval are frame-based, where the "shot" is the basic unit for indexing. Automatic or semi-automatic methods for temporal segmentation of video into shots and stories have been intensively studied [1]. In many cases, the visual content of each shot is summarized by a representative frame and its global features, such as a color histogram [2], [3]. An example of interactive summarization methods is the Scene Transition Graph (STG) [4]. STG is a directed graph that compactly captures both the image content and the temporal flow of the video. Each node in the graph represents a category of video shots, and each edge denotes a temporal relationship between them. However, temporal activities and events within a shot or a scene cannot be well represented by these frame-based approaches. It is generally agreed that humans analyze a video in terms of the objects of interest and their motions, where an object refers to a meaningful spatial/temporal region of a video. In other words, imagery data is processed by the human visual system to identify objects of interest, and then characterized in terms of these objects, their spatial and temporal properties and their interactions. To this effect, we adopt an object-based video description and indexing approach in this paper.

Object-based indexing requires objects to be identified and segmented first. Low-level attributes (shape, color, texture and motion) and other content information (object identity and semantics) are then attached to each object. One approach to object-based video description and indexing is to segment a video into shots and then select one or more key frames for each shot. Then, objects of interest are identified in each key frame, and the video is described and indexed in terms of the identified objects [5]. However, this approach treats a video source as a set of still images and thus ignores the time-variant or dynamic characteristics of the objects, such as variation in object's shape and the interactions of an object with other objects. Besides spatial features such as color, texture, shape and spatial composition, temporal features such as object motion, variation of object shape and interaction between multiple objects are key features to represent video content. Netra-V is an object/region-based video indexing system, which employs affine motion representation for each region [6]. Motion is a key object attribute in VideoQ [7], where regions that have similar motion trajectories can be retrieved. Our previous work in this area has considered 2-D mesh representation of video objects for indexing purposes [8]. These approaches are limited in that the temporal characterization of objects are only in terms of low-level motion characteristics. Other time variant features, such as object interactions and temporal segmentation of object motion to facilitate semantic-level description have not been considered. The notion of minimum description length segments was recently proposed to segment multiple moving objects [9]. It is based on finding spatially stable edge segments and describing their coherent motions with the most efficient motion model.

This paper introduces 1) a novel way to describe object motion by means of elementary motion units (EMU), action units (AU),

Manuscript received June 8, 1999; revised November 4, 2001. This work was supported in part by a National Science Foundation IUCRC grant and a New York State Science and Technology Foundation grant to the Center for Electronic Imaging Systems at the University of Rochester, and a grant by Eastman Kodak Company.

Y. Fu, deceased, was with the Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY 14627-0126 USA.

A. Ekin, and A. M. Tekalp are with the Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY 14627-0126 USA (e-mail: tekalp@ece.rochester.edu; mtekalp@ku.edu.tr).

R. Mehrotra is with the Imaging Science Division, Eastman Kodak Company, Rochester, NY 14650-1816 USA.

Publisher Item Identifier S 1057-7149(02)00804-7.

elementary reaction units (ERU), and interaction units (IU) and 2) a novel way to use dominant affine motion parameters to segment the lifespan of a video object into EMUs. An EMU is a set of consecutive frames within which the dominant motion of the object can be represented by a single parametric model. An ERU is a set of consecutive frames within which two video objects have a predefined interaction. An AU is defined as a time-ordered sequence of EMUs, while an IU is that of ERUs. The concept of the proposed hierarchical object-based motion description for video is described in Section II, where we also introduce an object motion/interaction description graph that provides a visual summary of the video in terms of low-level EMUs and ERUs, high-level AUs, and IUs. In Section III, we propose an algorithm that automatically segments the life-span of a video object into EMUs, and also computes a representative (dominant) affine model for each EMU. In the next section, we provide algorithms for the identification of ERUs and the classification of types of ERUs. Experimental results for EMU segmentation and applications of the proposed methods to video browsing and EMU retrieval are presented in Section V.

II. Hierarchical Object-Based Motion Description

This section presents the concept of the proposed hierarchical object-based motion description of video. We define an object-based segment as a selected occurrence of a set of objects between a begin frame and an end frame. In the proposed framework, the static content of an object-based segment consists of one or more foreground objects and the corresponding background object(s). The motion of each object and a set of object-to-object interactions describe the dynamic content of the segment. We propose a two-level hierarchical—a low-level and a high level-description for both object motion and interactions. The complete framework for the proposed hierarchical object-based video description scheme is depicted in Fig. 1.

The proposed framework is based on the following observations: 1) Low-level motion of the objects and their interactions are generally too complex to describe at the segment level; thus, the need to segment the motion into smaller temporal units, called EMU and ERU. To further simplify object motion description, we only consider the dominant motion of each foreground object. 2) Physical motions exhibit temporal continuity due to the inertia of the motion; hence, there is a strong correlation between the parameter vectors describing the dominant motion between the successive pairs of frames. In order to exploit the redundancy in the sequence of dominant motions of a foreground object between successive pairs of frames, we assign a single representative dominant parametric motion model for each EMU and a single interaction type for each ERU. Thus, an EMU is defined as a set of consecutive frames, within which the dominant motion of the object can be represented by a single parametric model, and an ERU is a set of consecutive frames, within which two video objects have a predefined type of interaction. In Section III, we propose an algorithm for the automatic segmentation of the motion of a video object into low-level EMUs based on affine motion modeling, and we compute a representative affine model for each EMU. We define three types of spatio-temporal relationships to identify ERUs. They are based on

1) object boundaries
2) object positions
3) object motions

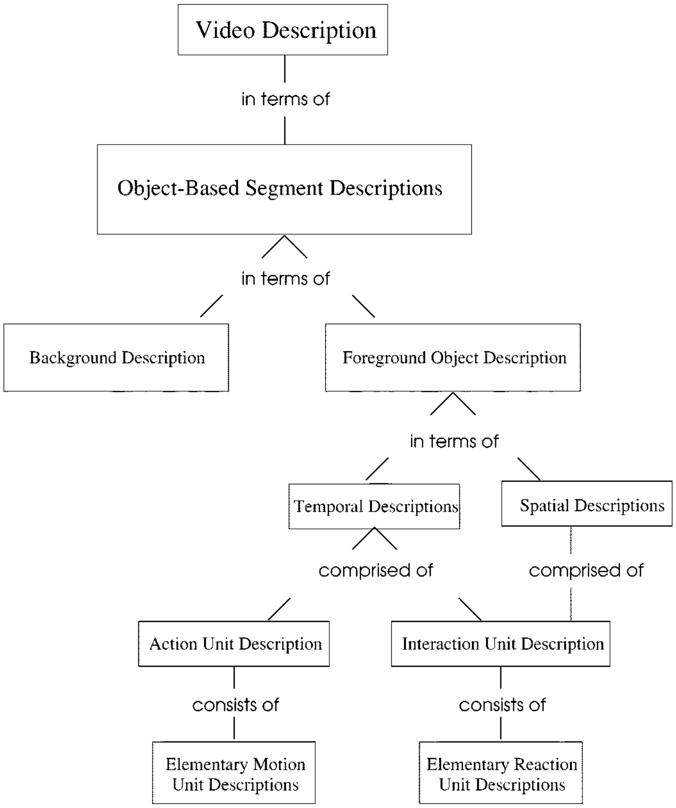


Fig. 1. Overview of the hierarchical object motion/interaction description scheme.

Automatic extraction of ERUs is discussed in Section IV. At this stage, we can retrieve EMUs and ERUs from a database based on similarity of representative dominant affine motions and low-level interaction types, respectively. Next, we employ the identified EMUs and ERUs to facilitate semantic level description of motion.

Humans interpret and describe motion at the semantic level rather than in terms of low-level features. To this effect, we define an action unit (AU) as a time-ordered sequence of EMUs of an object, which carries a semantic meaning. Therefore, EMUs focus on the low level motion coherence, while AUs focus on semantic abstraction. For example, the action of throwing may consist of a sequence of EMUs, which correspond to different states of the arm and body during the throwing motion. Similarly, an ordered set of consecutive ERUs corresponding to a meaningful semantic interaction between two objects forms a semantic-level unit for the description of object's interaction, called an interaction unit (IU). Here we assume that the grouping of EMUs into AUs and ERUs into IUs is done interactively. The automatic mapping of low-level motion features into semantic description is beyond the scope of this paper. We developed a graphical user interface, whereby a user can browse automatically detected EMUs and ERUs, combine them into AUs and IUs, and annotate them with text strings, interactively. This

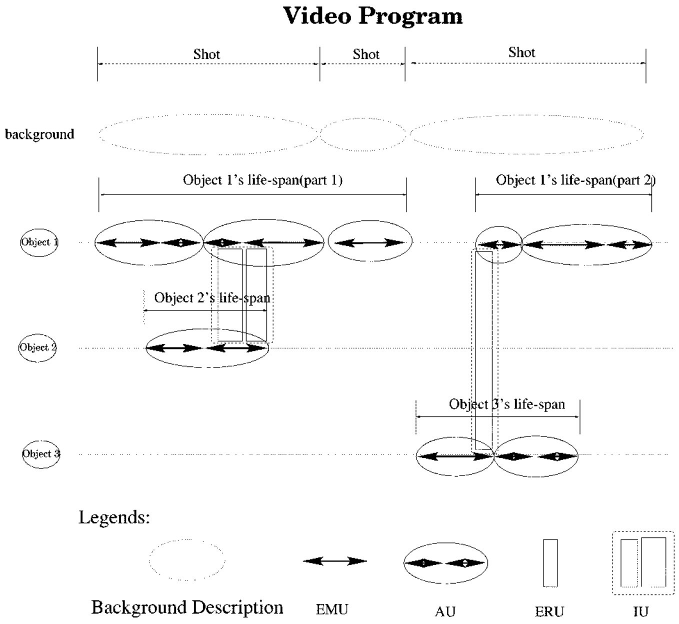


Fig. 2. Object motion/interaction description graph.

tool facilitates the generation of an object motion/interaction description graph with semantic annotation for visual summarization.

The object motion/interaction description graph is a means of visual summary of the object-based motion description of a video. A template of this graph representation for a generic video program is depicted in Fig. 2, where the life-span of each object consists of one or more shots. Each shot has a separate background object, while a foreground object may appear in more than one shot. At present, the linking of objects across shots into a life-span is handled interactively. Both low-level and semantic-level temporal units of motion of each object as well as interactions between objects are summarized over the life-span of each object. We demonstrate the object motion/interaction description graph for video browsing applications in Section V-C.

An overview of the method to compute the proposed description of a video is outlined in Fig. 3. The first step is to detect or select occurrences of the objects of interest in the video. For certain applications, a precompiled set of object models can be used to automatically detect the occurrences of objects by model-based or appearance-based object detection [10]. Alternatively, if the video is chroma-keyed and/or encoded using an object-oriented scheme such as MPEG-4, then this information is contained in the alpha plane of each object. Another option is to employ user interaction, where the object of interest is manually identified in the first frame and then automatically tracked until the object ceases to exist [11]–[13]. We detect an occurrence of an object by detecting shots and searching each shot to define a new occurrence of the object. In our implementation, shot boundary detection is performed using the method proposed in [14]. The objects of interest are manually identified in the first frame of each shot. After identifying the objects of interest, the segments corresponding to the life-span of each object are computed using the occlusion-adaptive motion snake [12] and identified by the begin and end frame for the objects appearance. Next, the life-span segment of the object of interest is partitioned into EMUs and ERUs automatically (as described in the following sections), and appropriate descriptors for each are computed. Finally, EMUs and ERUs are grouped into semantically meaningful AUs and IUs, respectively.

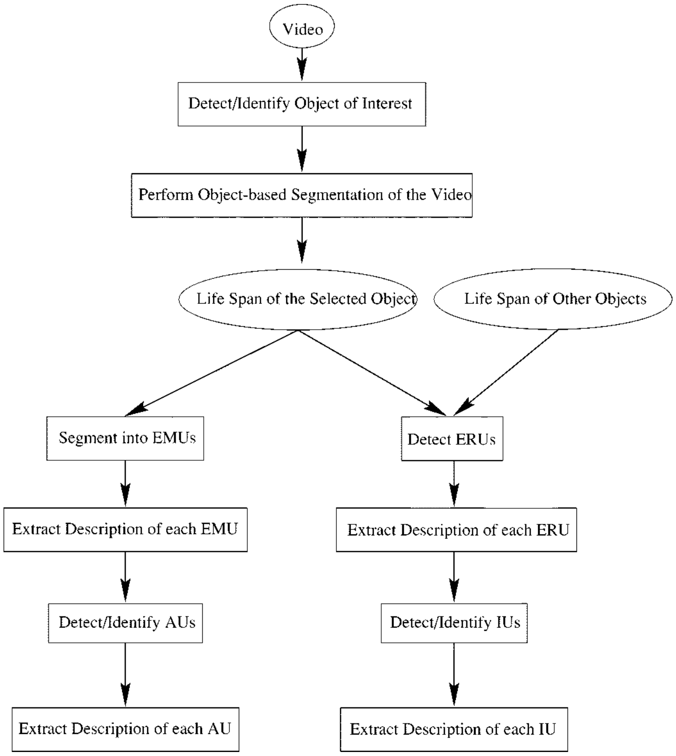

Video
Detect/Identify Object of Interest
Perform Object-based Segmentation of the Video
Life Span of the Selected Object
Life Span of Other Objects
Segment into EMUs
Detect ERUs
Extract Description of each EMU
Extract Description of each ERU
Detect/Identify AUs
Detect/Identify IUs
Extract Description of each AU
Extract Description of each IU

Fig. 3. Extraction of the proposed hierarchical object-based motion description scheme.

III. Temporal Segmentation and Description of Object Motion

This section addresses the segmentation of the foreground object motion into EMUs, and, in Section III-A, description of each EMU. We discuss compensation for and indexing of the background motion in Section III-B, although background motion compensation is estimated first in order to access the actual foreground object motions (and not relative motions with respect to the camera).

A. Low-Level Object Motion Description: Elementary Motion Units

We define an EMU as a set of consecutive frames within which the dominant motion of the object remains more or less the same, i.e., it can be approximated by a single parametric motion model. Within each EMU, only one vector of parameters, the representative parametric motion model, representing the dominant motion of the EMU is kept. Then, the motion of an object within its life-span can be approximated by a sequence of parametric motions, one for each EMU, and by other descriptors, if necessary. That is, the EMU is the atomic temporal unit in our motion description. The number of EMUs used to represent the life-span of an object is determined by the object motion and the required precision of the description. The following key steps are used to segment the life-span of each object into EMUs:

1) Parametric Model Fitting Between Successive Pairs of Frames: The 2-D motion field between each adjacent pair of frames within the object's lifetime in a shot is described by fitting a parametric motion model. There are a number of motion models [15], e.g., 2-D translation, translation-rotation-zooming and the 2-D affine model. In this paper, a six-parameter affine motion model is employed to describe an object's dominant motion. As pointed out in [16], affine motion model is capable of describing 2-D translation, rotation, magnification and shear. The affine model is defined by:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = A \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \qquad (1)$$

where a_i, $i = 1, \ldots, 6$, in the matrix, A, are the model parameters. The transformation displaces a point $(x, y)^t$ in the first frame (the reference frame) to a new location $(x', y')^t$ in the second frame (the target frame). The origin of the image coordinates in the reference frame is used as the spatial reference for the transformation. A least square based algorithm is used to extract the motion model from the dense motion field [17], such that

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \left(\begin{bmatrix} \sum x \\ \sum y \\ 1 \end{bmatrix} \begin{bmatrix} \sum x & \sum y & 1 \end{bmatrix} \right)^{-1} \begin{bmatrix} \sum xx' \\ \sum yx' \\ \sum x' \end{bmatrix} \qquad (2)$$

and

$$\begin{bmatrix} a_4 \\ a_5 \\ a_6 \end{bmatrix} = \left(\begin{bmatrix} \sum x \\ \sum y \\ 1 \end{bmatrix} \begin{bmatrix} \sum x & \sum y & 1 \end{bmatrix} \right)^{-1} \begin{bmatrix} \sum xy' \\ \sum yy' \\ \sum y' \end{bmatrix} \qquad (3)$$

where all summations are performed over the object region. Procedures to eliminate outlying dense motion vectors are employed to improve the accuracy of the representation.

To describe how well the model agrees with the data, we compute a confidence measure, which is a weighted sum of the motion vector and motion-compensated intensity mismatch within the object boundary, as follows:

$$\begin{aligned} CON_p = {} & \alpha_1 \sum_{(x,\,y) \in object} \|V_r(x, y) - V_m(x, y)\|_2 \\ & + \alpha_2 \sum_{(x,\,y) \in object} \|C_r(x, y) - \text{Warp}(C_t, A, x, y)\|_2 \end{aligned} \qquad (4)$$

where α_1 and α_2 are constants, $V_r(x, y)$ and $V_m(x, y)$ are the dense and parametric motion vectors at $(x, y)^t$, respectively; $C_r(x, y)$ and $C_t(x, y)$ are the color vectors of the image pixel at location $(x, y)^t$ in the reference and target frames, respectively. $\|\cdot\|_2$ stands for L_2 norm. The warping operator is defined as

$$\text{Warp}(C_t, A, x, y) = C_t(x + u, y + v) \qquad (5)$$

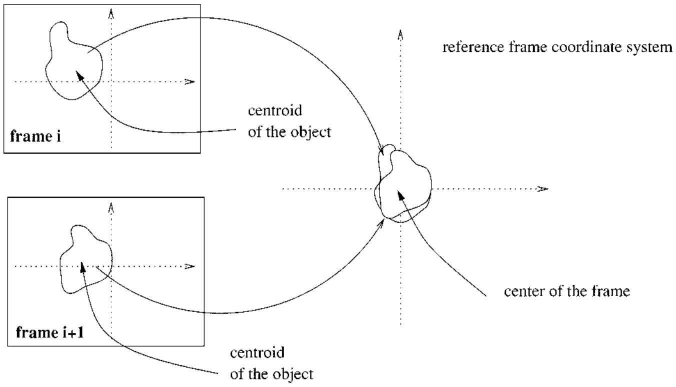


Fig. 4. Extraction of the proposed hierarchical object-based motion description scheme.

where $(u, v)^t$ is the motion vector at $(x, y)^t$ in the reference frame:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} - \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}. \qquad (6)$$

The smaller the value of CON_p, the better the fit of the parametric model. The value of CON_p between a pair of frames in (4) is subsequently used to determine how much confidence can be associated with the EMU boundaries computed in Section III-A2 below.

2) Computation of Dissimilarity Measure: The initial set of elementary motion units is obtained by identifying frames corresponding to a significant change in the object's dominant motion. This is accomplished by computing the dissimilarity of the estimated motions between adjacent frame pairs and by identifying the local maxima in the frame-by-frame dissimilarity function. Several researchers [18] have indicated that the distance between two sets A_1 and A_2 of motion parameter vectors is not a meaningful way to measure the visual similarity of the resulting motion fields. Therefore, we propose a dissimilarity measure between two parametric motion models in terms of the dense motion fields computed by these models through the following steps.

1) Align centroids of two objects in the two frames to the centroid of the image coordinate (see Fig. 4), resulting in two new parameter sets A'_1 and A'_2.
2) Calculate the following dissimilarity measure

$$DSIM(A'_1, A'_2) = \frac{\sum_{(x,\,y)} M(x, y) \|V_1(x, y) - V_2(x, y)\|_2}{\sum_{(x,\,y)} M(x, y)} \qquad (7)$$

where the weight factor is defined as

$$M(x, y) = \begin{cases} 2, & \text{if two object masks overlap at } (x, y) \\ 1, & \text{if } (x, y) \text{ is in only one object mask} \\ 0, & \text{if } (x, y) \text{ is in neither object masks} \end{cases} \qquad (8)$$

reflecting that dissimilarity between motion fields of the two objects is determined by the overlapping regions.

The motion vector $V_i(x, y)$ generated by the affine model A_i at location $(x, y)^t$ is as follows:

$$V_i(x, y) = \begin{bmatrix} u_i(x, y) \\ v_i(x, y) \end{bmatrix} \qquad i = 1, 2$$

$$\begin{bmatrix} u_i(x, y) \\ v_i(x, y) \\ 1 \end{bmatrix} = A_i \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} - \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}. \tag{9}$$

An adaptive dissimilarity threshold, Q, calculated as half of the maximum frame-to-frame dissimilarity measure (7) over the whole sequence, is employed such that frames having dissimilarity measure greater than Q are selected as the initial set of EMU boundaries. If the motion dissimilarity measure between the first and last frames of any selected EMU is above the threshold Q, then that EMU is divided in the middle into two EMUs. The process is repeated until no more splitting can be done. Next, two adjacent EMUs are merged if the motion dissimilarity between the middle frames of two EMUs is less than or equal to the threshold Q. This yields the final set of EMUs of an object. We also associate a confidence measure CON_b with the detected EMU boundary locations, which can be computed as the average value of the pairwise confidence measures CON_p in a local neighborhood of the detected EMU boundary location. This confidence measure may be used to manually check or correct EMU boundary locations with low confidence values if desired.

3) Computation of the Representative Motion Model: For each EMU, one parametric motion model is kept as a representative model. To select an affine model that best represents the dominant motion within an EMU, we consider the following requirements. 1) The representative model should be one of the affine models within the EMU (since, for example, averaging the component model parameters is generally not physically meaningful). 2) The selection process should be robust to outliers within the EMU. We propose to select the median of the motion parameter sets, between pairs of frames within an EMU, as the representative motion model in order to satisfy these conditions.

The vector median $\underline{a}_{MED}$ of a set of vectors $W = \{\underline{a}_j j = 1 \cdots N\}$, where N denotes the number of frames within an EMU, consisting of N 6-D affine motion parameter sets, $\underline{a}_j = [a_{j1}, a_{j2}, a_{j3}, a_{j4}, a_{j5}, a_{j6}]^t$, is defined as [19], [20]

$$\underline{a}_{MED} = \arg \min_{\underline{x}_j \in W} \sum_i \|\underline{x}_i - \underline{x}_j\|. \tag{10}$$

Motion models of frames that are close to the EMU boundaries are discarded since they may correspond to transition frames between two EMUs. This is achieved by making two passes over each EMU, where outliers are eliminated in the second pass. Finally, a confidence measure CON_r can be associated with this representative parametric model. It can be computed as the average value of CON_p for the frames used in the computation of the vector median $\underline{a}_{MED}$ (after eliminating the outliers). An example of the application of CON_r may be that a segment with a high value of CON_r may not be a good candidate as a query for retrieval.

In addition to a quantitative low-level description of the object motion within an EMU (as shown previously), we also construct a visual representative for an EMU automatically for fast browsing and video summary applications. A thumbnail visual representative of an EMU is an image, obtained by overlaying the first, middle, and last frames in the EMU. The trajectory of the centroid of the object is also marked on this image. Therefore, an EMU E is described by the begin and end frame numbers, the representative dominant affine motion parameters, the trajectory of the object centroid within the EMU, and a thumbnail visual representative of the EMU.

B. Compensation and Indexing of Background Motion

Basic camera motion, such as camera pan or zoom, is very common in video production. We first perform camera motion compensation to recover the absolute motion of foreground objects. An appropriate parametric motion model is selected to represent the background (camera) motion for each shot. For indexing purposes, we employ a semantic annotation, which identifies the dominant camera motion within the shot, e.g., pan, tilt and zoom. To achieve these goals, we used a variation of the automatic dominant camera motion annotation method described in [21], which is summarized below for completeness.

First, a singularity test is performed on the optical motion field by comparing the mean and variance of the magnitudes of $((u(x, y))/x, (v(x, y))/y)^t$ for the background motion field as follows.

1) If the nonsingular components dominate, the background motion is due to camera rotation (pan, tilt) or translation (horizontal/vertical). Then, if the variance of the magnitude of the background motion vectors is greater than a threshold, the observed background motion is due to camera rotation; otherwise, it is due to camera translation.
2) If the singular components dominate, z-rotation, z-translation, or zooming is more important. A 2-D affine model is estimated from the background motion field

$$\begin{aligned} \begin{bmatrix} u(x, y) \\ v(x, y) \end{bmatrix} &= A \begin{bmatrix} x \\ y \end{bmatrix} + b \\ &= \begin{bmatrix} a_1 & a_2 \\ a_4 & a_5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} a_3 \\ a_6 \end{bmatrix} \end{aligned} \tag{11}$$

where a_i, $i = 1, \ldots, 6$, are model parameters. The matrix A is projected along two orthogonal bases I_1 and I_2 defined by

$$I_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad I_2 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \tag{12}$$

If the projection along I_2 is greater than that along I_1, then the dominant motion is due to camera z-rotation; otherwise, it is due to either camera zoom or camera z-translation. (Estimation of the model parameters is discussed in Section III-A.)

IV. Identification and Description of Object Interactions

Another important component of the dynamic content of video is object interactions. Similar to object motion description, we introduce two levels to describe object-to-object interactions: elementary reaction units (ERU) and interaction units (IU). The interaction between objects within a shot can be described at the low-level by a sequence of ERUs, which can be interactively grouped together to form semantically meaningful IUs.

A. ERU Identification

We define three basic low-level interaction (reaction) types with various low-level interactions as follows:

- Reactions related to object boundaries
  - —Coexistence: Two objects appear together.
  - —Physical Contact: Two objects have spatial contact, i.e., their boundaries touch.
  - —Occlusion: One object occludes the other, i.e., their boundaries overlap.
- Reactions related to object positions (following [22], [23])
  - —Directional Relations: Strict directional relations (*north, south, west,* and *east*), mixed directional relations (*north-east, north-west, south-east,* and *south-west*), and positional relations (*above, below, top, left, right, in front of, behind, near,* and *far*).
  - —Topological Relations: *Equal, inside, disjoint, touch,* and *cover*.
- Reactions related to object motions
  - —Approach: Object 1 is approaching object 2.
  - —Diverge: Object 1 is moving away from object 2.
  - —Stationary: The objects are stationary with respect to each other.

The first step is to identify, at each frame, the type of low-level interaction between every pair of objects. Next, an ERU is detected by locating the set of consecutive frames having the same identified low-level interaction between the same pair of objects.

Automatic extraction of the above low-level interactions is simple given the alpha plane (segmentation maps) for each object. Coexistence is determined from the life-span of each object. If there is an overlap between the life-spans of the two objects, then their interaction is "coexistence." Spatial boundaries of two coexisting objects are checked to find a common boundary. Existence of a common boundary between the objects results in "contact" interaction, which is a special case of "coexistence." The change in the ratio of contacting objects' sizes determines the "occlusion." If the ratio increases in time then the first object occludes the second, if it decreases, the second occludes the first.

Spatial relationships, such as "right of," "left of," "above," and "below," are found from the locations of the object centroids, and bounding boxes. The "in front of," and "behind" relations require the recovery of depth information which is not applicable for some applications. Motion relationships are described using the relative motion between two objects. The algorithm divides each object into an equal number of small rectangular patches and selects the center of each rectangular region. Corresponding patch distances are calculated and updated using the motion parameters of each object. The difference in the first distance set and the second determines the motion relationship, which can be "approach," "diverge" or "stationary."

After finding each basic low-level interaction type, all consecutive frames with the same type are merged to result in the final ERU segments. The algorithm finds different ERUs for each basic type, resulting in a total number of ERUs that is linear in the number of basic types. Note that all ERUs are classified into three classes (object boundary, position, and motion).

B. ERU Description

An ERU is described by two object identifiers, start and end frame numbers, the interaction type, and the interaction-specific descriptors. We define the following interaction specific descriptors:

1) Coexistence is described by the first and last frames of the overlapping time period.
2) Physical contact is described by the boundary of contact in addition to the first and last frames of the contact period.
3) Occlusion is described by the occlusion ordering of the two objects in addition to the description of the occlusion boundary and first and last frames of the occlusion period. Occlusion by other objects can also be detected.
4) Position is given as the name of relationship and spatial distance of objects.
5) Motion is described by the name of the relationship and the relative speed of objects.

V. Applications and Experimental Results

We present three applications of the proposed hierarchical description of object motion and interactions: video segmentation, EMU retrieval and video summarization. Experimental results for automatically segmenting a video sequence into EMUs and ERUs are presented in Section V-A. The representative affine motion parameter set of an EMU is used as a query feature in Section V-B for EMU indexing and retrieval. Finally, the visual content of video sequences is summarized at the semantic level by an object motion/interaction description graph for a browsing application in Section V-C. The following video sequences, with a total of 2600 frames, are used in our experimental results.

1) Seven "Indoor" sequences were captured and digitized at 30 frames per second in our laboratory. The number of frames in these sequences are 169, 154, 329, 305, 138, 375 and 179, respectively, where each frame has a resolution of 320 × 240. In each of these sequences, a different person (foreground object) appears and performs simple motions, such as walking, sitting down, standing up, etc. A green chroma-key was used as the background to enable easy object segmentation.
2) The "Children" sequence is an MPEG-4 test sequence. There are 300 frames, each of which is 352 × 288. The sequence shows two boys playing with a ball. One picks up the ball and throws it to the other. Object segmentation

TABLE I
EMU BOUNDARY DETECTION RESULTS OF TEST SEQUENCES. NOTE: $*$ INDICATES A FALSE ALARM AND ’ INDICATES A MISS

Sequence	Begin/End Frames	EMU Boundary Frame Numbers
Indoor 1: yufuw	0 - 168	8*, 19(14-25), 45(35-46), 63(57-59), (92-93)’, 135*, (127-130)’, 149*
Indoor 2: yufus	0 - 329	(9-12)’, (32-37)’, 54(54-61), 84(83-92), 123(122-124), (165-166)’, 194(194-202), 244(238-244), 255*(261-271)’, 291(291-297)
Indoor 3: yaxuw	0 - 154	19, 37(32-38), 52(47-56), 70(64-77), (100-108)’, 142(139-143), (150-153)’
Indoor 4: yaxus	0 - 305	6*, 24*, (59-63)’,80*, 97(91-98), 138(129-142), 151(157-162)’, 187(177-188), (206-212)’, (228-230)’, (264-271)’, 287*
Indoor 5: lishaw	0 - 138	22(16-25), 44(43-48), (65-69)’, (87-89)’, 112(111-118), 130*
Indoor 6: lishas	0 - 375	30(26-33), 57(45-55), 78(80-84), 157(151-153), 185(181-191), 215*(208-212)’, 229(230-235), 258(255-260), 306(309-315), 335(338-344), 355(352-354), 369*
Indoor 7: mugew	0 - 179	7*, 21(19-28), 57(54-59), 131(124-133), 145(144-145), 161(154-164)
Children: Boy on the left	0 - 299	(18-22)’, 26(28-31), (40-42)’, (80-84)’, 92(88-94), 107(102-114), 143*, 181(180-185), 201(196-202), 219(216-227), 254(252-257), 288(289-295)
Children: Boy on the right	0 - 299	(9-13)’, 23(20-23), 41*, (46-50)’, (72-74)’ ,82(78-82), 115(110-114), 126(124-127), 171(170-175), 201(202-205), 229(224-225), 254(252-256), 267(260-266), (270-273)’
Playboy : Young man	2530 - 2719	2537(2537), 2600(2589-2599), 2622*, 2637(2630-2636), 2657(2652-2662), 2669*, (2676-2679), 2685*, (2693-2702)’
Playboy : Lady	2720-2820	2730*, 2768*

TABLE II
PERFORMANCE EVALUATION OF EMU BOUNDARY DETECTION

	Indoor							Children		Playboy	
Recall rate	3/5	6/10	4/6	3/8	3/5	10/11	5/5	8/12	9/14	4/6	0/0
Precision rate	3/6	6/7	4/5	3/8	3/4	10/12	5/6	8/9	9/10	4/7	0/2

of each boy and the ball were generated for every frame using our snake-based semi-automatic object segmentation tool [12].

3) The “Playboy” sequence is an MPEG-7 test sequence. We chose 957 frames (frame 2530–3486) including four shots (2530–2719, 2720–2820, 2821–3202, 3202–3486). Each frame is 352 × 288. The sequence shows a young man entering a room where a lady is writing a letter on a table. He closes the door, takes off his hat and talks to the lady.

A. Temporal Segmentation of Video into EMU and ERU

The simplest sequences to process are the “Indoor” sequences, where only one foreground object appears in each sequence; thus, the life-span of the foreground object is the entire sequence. There are three foreground objects in the “Children” sequence: the boy on the left, the boy on the right and the ball. They appear in the entire sequence; thus, the life-span of each object is again the entire sequence. The life-span of each boy is segmented into EMUs according to the dominant affine motion. In the “Playboy” sequence, there are a total of two foreground objects, but only one appears in each of the manually-selected four shots. Each shot is segmented into EMUs according to the dominant motion of that foreground object in that shot. Table I lists the automatically detected EMU boundaries versus the ground truth ranges which are identified by subjective tests for each sequence. Each sequence is viewed by four viewers (in frame-by-frame mode), who were asked to identify the visual breaks in the dominant motion of the foreground object. The results were compiled to form the ground truth ranges for the breaks in the dominant motion. In Table I, automatically detected results are listed in temporal order separated by commas, with ground truth ranges placed in parentheses next to the corresponding detection results. A ground truth range with no numerical value next to it refers to a miss. A numerical value listed without a corresponding ground truth range denotes a false positive.

The performance of the EMU boundary detection method, discussed in Section III-A, on the test sequences is summarized in Table II. Recall rate is the number of correct returns (i.e., detected EMU boundaries that fall within the ground truth range as indicated in Table I) divided by the total number of desired items (ground truth EMU boundaries within the life-span of the object). Precision is the number of correct returns divided by the

TABLE III
ERU DETECTION RESULTS 1 (APP. = APPROACH, DIV. = DIVERGE, ST. = STATIONARY)

	Boy on the Left											
ERU Begin End Frames	0 20	21 23	24 79		80 82	83 272					273 292	293 299
ERU (boundary)	CE	OC	CE		CT	CE					OC	CT
ERU(position)	Left											
ERU Begin End Frames	0 23	24 42	43 70	71 79	80 122	123 155	156 170	171 220	221 250	251 272	273 292	293 299
ERU(motion)	App.	Div.	St.	App.	Div.	St.	App.	St.	Div.	App.	Div.	App.

TABLE IV
ERU DETECTION RESULTS 2. ACRONYMS: CE = COEXISTENCE, CT = CONTACT, OC = OCCLUSION

	Boy on the Right											
ERU Begin End Frames	0 6	7 12	13 38	39 69	70 75	76 105	106 163	164 174	175 229	230 240	241 252	253 299
ERU(boundary)	OC	CT	CE	OC	CT	CE	OC	CT	CE	OC	CT	CE
ERU(position)	Right											
ERU Begin End Frames	0 23		24 42	43 70	71 79	80 122	123 155	156 170	171 220	221 250	251 272	273 299
ERU(motion)	Div.		App.	St.	Div.	App.	St.	Div.	St.	App.	Div.	St.

total number of returns (including false positives). Note that for the boy on the left in the "Children" sequence, the method failed to detect several EMU boundary locations identified by a user, e.g., when the boy raises his hands. This is due to the fact that the sequence is segmented according to the object's dominant motion. When there is a sudden change of motion of only a small part of the object, the algorithm ignores it as an outlier. Another possible reason for the errors in the experiment is that only a global threshold is used.

Low-level object-to-object (including background object) spatio–temporal relationships are detected using the method described in Section IV. For the Indoor and the Playboy sequences, there are no ERUs since only one object appears in each shot. The reactions between the two boys in the "Children" sequence do not change in the entire sequence: coexistence, left (with respect to the boy on the left), disjoint, and stationary. The reactions between the boy on the left and the ball, and the boy on the right and the ball are given in Tables III and IV, respectively. The first two rows in each table show the ERU boundary locations for the reaction type "object boundary." The ERU boundaries for reaction types "object position (positional reaction only)," and "object motion" are also included. An example of spatial and motion-related interactions can be seen in Fig. 5 which shows a single representative frame for several ERUs, such as "The ball is diverging from, to the left of, has no contact with the boy on the left," and "The ball is approaching, to the right of, has no contact with the boy on the right." The evaluation of the ERU boundaries is straightforward given the alpha planes of each object in the sequence. Errors in detecting ERUs may arise if there are errors in the alpha planes. Since we use semi-automatic object segmentation, this does not occur in our results.

B. EMU Retrieval

In this section, we present two experiments with video indexing/retrieval by object motion. In particular, we select an EMU as the query EMU and consider retrieval of all EMUs in an EMU database with similar motion content. In both experiments, we use the representative affine motion model to describe the motion of each EMU. The retrieval results are ranked in increasing order of dissimilarity measure, where the dissimilarity between the representative affine models of two EMUs is computed by three different dissimilarity metrics

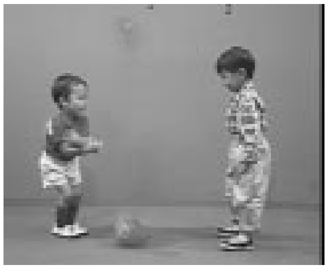


Fig. 5. Single representative frame for ERUs of objects in Children sequence.

$$DSIM = \frac{CON_r . \sum_{(x,y)} (w(x,y) \| V_1(x,y) - V_2(x,y) \|_2)}{\sum_{(x,y)} w(x,y)} \tag{13}$$

$$DSIM = \frac{CON_r . \sum_{(x,y)} (w(x,y) \| V_1(x,y) - V_2(x,y) \|_1)}{\sum_{(x,y)} w(x,y)} \tag{14}$$

$$DSIM = CON_r . \sum_{n=1}^{6} w_n |a_{1n} - a_{2n}|. \tag{15}$$

In (13) and (14), object centroids are translated to a new origin that corresponds to each object's center of gravity. Velocities are calculated using the dominant affine parameters multiplied by a

TABLE V
RECALL AND PRECISION PERCENTAGES FOR QUERIES IN THE CAPTURED TEST SEQUENCES

Number of Returned Items	Query 1	Query 2	Query 3	Query 4	Query 5
1	*14/100*	*25/100*	*17/100*	*10/100*	*6/100*
3	*43/100*	*50/67*	*50/100*	*30/100*	*19/100*
5	*71/100*	50/40	*67/80*	*50/100*	*31/100*
7	86/86	50/29	67/57	*70/100*	*44/100*
10	86/60	50/20	67/40	80/80	*62/100*

weight factor described in the same way as (7) while w_ns in (15) are the weights for affine parameters that are taken appropriate for the dynamic range of the parameters. CON_r is the measure, explained in Section III-A3, to penalize EMUs with low confidence in representative affine parameters. A small CON_r value, i.e., a small difference between the affine set and the 2-D dense motion field, for an EMU means a better fit of the affine parameters to the dense motion fields of frames constructing that EMU.

In the first experiment, we consider 64 EMUs from all seven Indoor sequences simultaneously. Given a query EMU, we wish to retrieve all EMUs with similar motion content using affine metrics from the combined EMU database. Table V shows the recall and precision rates for the captured test sequences. The recall rate is the number of desired items returned by the query divided by the total number of desired items in the database. Precision is the number of desired items returned by the query divided by the total number of the items returned (including false positives). We limit the number of returned items to 1, 3, 5, 7, and 10 and show the recall and precision rates in percentages. In the second query, the precision rates are not satisfactory due to the large number of close EMUs in terms of dominant motion that affects only a small region of the body and/or due to a possible motion estimation error.

Another experiment using the "Children" sequence gives results using a real scenario. There are 15 EMUs for the boy on the right and 13 EMUs for the boy on the left. We provide a user-friendly graphical interface where the user selects the query image and the metrics and the closest ten matches are shown in another window. The results of the retrieval are shown for affine parameters and L_2 velocity field matching (L_1 velocity matching did not generate better results so we skip its discussion here) in Table VI. We conclude that the query is returned correctly if the subjective results appear among the best five matches replacing recall/precision rate. Velocity field matching resulted in poorer query returns than affine metrics in this experiment since our queries included nonrigid motion where dominant motion, possibly caused by moving hands, is spread to the whole object area in the velocity calculation. A sample query for the third EMU of the boy on the right is shown in Fig. 6. In summary, the high rate of correct returns in the first and second experiments is satisfactory for our purposes. Velocity metrics can be improved by adjusting weights automatically to favor the high motion areas rather than calculating velocity in the whole object region.

C. Visual Summarization by Object Motion/Interaction Description Graph

The object motion/interaction graph provides an object-based visual summary of a video sequence. The graph for the Children sequence is shown in Fig. 7. A list of the three foreground objects that appear in the sequence are shown along the vertical axis. An object icon and a set of spatial (static) features (e.g., color, shape, etc.) can be associated with each object. The duration of shots and the life-span of each object is depicted along the horizontal axis. In this case, there is only one shot whose duration is equal to the life-span of all three objects.

TABLE VI
EMU RETRIEVAL RESULTS

Query no	Affine parameters Matched/Expected	L_2 Vel. field Matched/Expected
1	3/3	3/3
2	3/3	3/3
3	2/2	2/2
4	2/2	1/2
5	3/3	2/3
6	3/3	3/3
7	3/3	3/3
8	3/3	3/3
9	1/2	0/2
10	1/1	1/1
Total	24/25	21/25

At the AU/IU level, the object life-span is a single-branch graph tree of time-ordered AUs. The object's AUs are presented by their visual summaries enclosed in a box. A user can click on the box and a pop-up window will display all the associated EMU visual summaries in temporal order. An IU is the connection (the rectangle that encloses one or more ERUs) among object life-spans. A click on the IU will open a window listing all associated ERUs. Semantically related AUs and IUs can be grouped into a story interactively.

At the EMU/ERU level, an object's life-span is a single-branch tree of time-ordered EMUs, grouped into AUs. Each EMU is represented by its visual summary enclosed in a box. A user can click on the box to view the corresponding original video segment. An ERU is presented as a connecting rectangle between two objects, annotated by the low-level spatio-temporal relation type.

When browsing a video using the proposed object motion/interaction graph, a user can access the video sequence through either individual objects or through shots. A user can select the object's icon from the object's list and scan the object life-span at AU/IU level to find an interesting action and/or interaction through related EMUs and ERUs. The user can jump to other objects in the IUs and ERUs in which the current object participates. The other way to browse a video is to go through a shot/story. The user first browses the story list and selects an in-

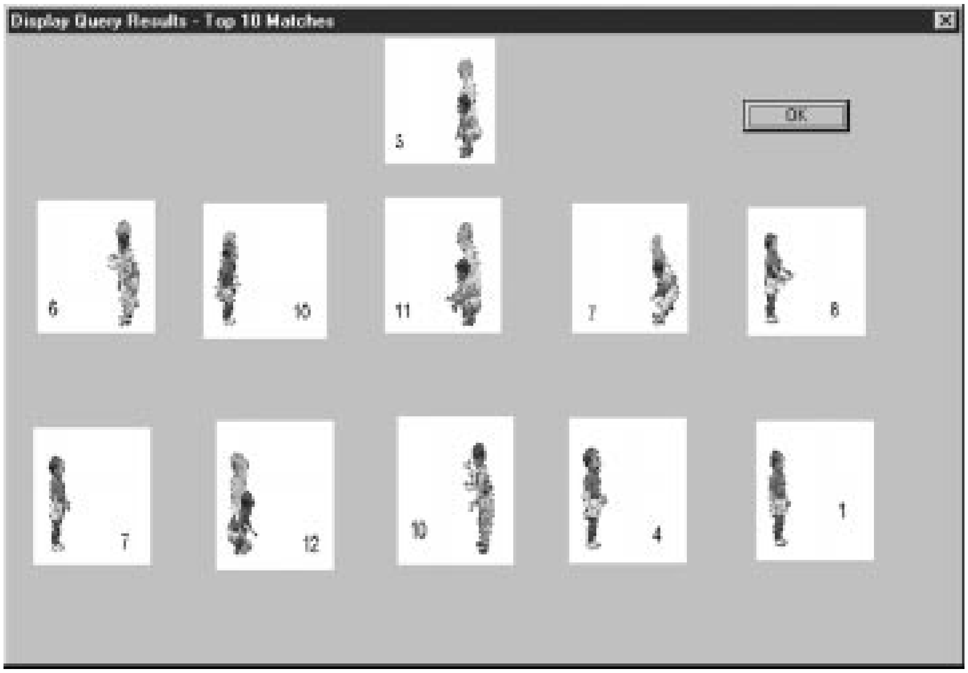


Fig. 6. Query result for the third EMU of the boy on the right.

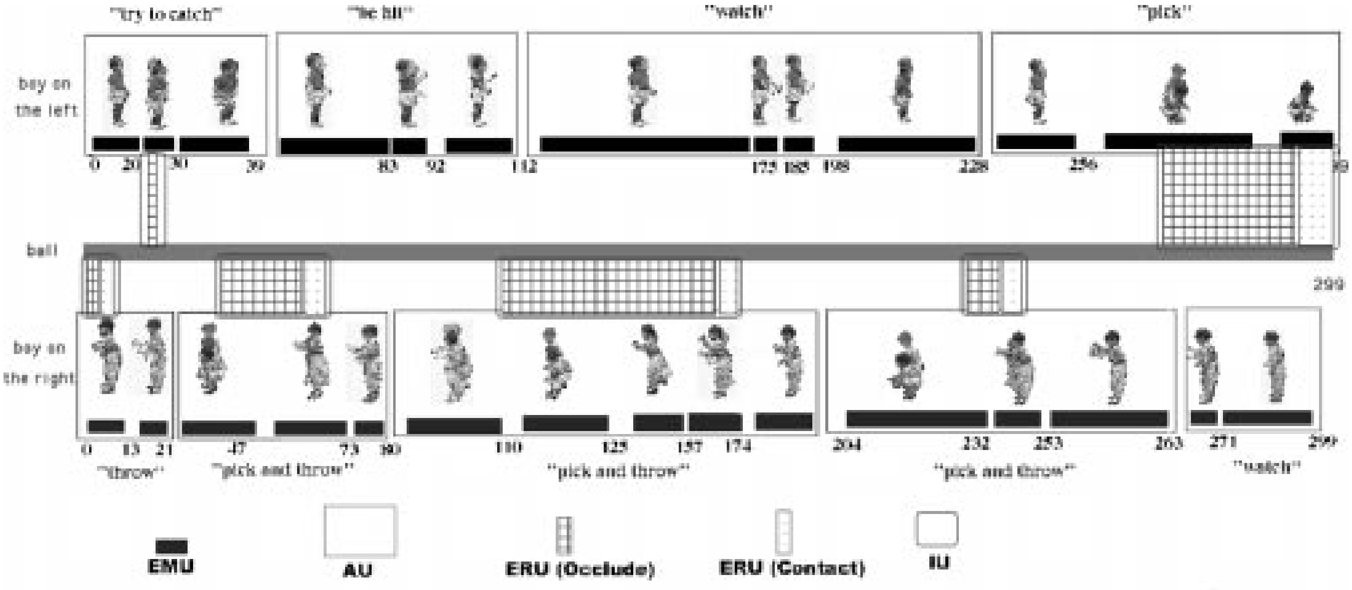


Fig. 7. Object motion/interaction description graph (the annotations in quotation marks are manually determined).

teresting story, then the user can view each object in that shot and view its AUs and EMUs in its life-span.

VI. Conclusions

In this paper, we have described an object-based video description hierarchy, in terms of EMUs and AUs, and ERUs and IUs for object motion and interaction, respectively. Automatic algorithms that extract the low-level elements, i.e., EMUs and ERUs, and their descriptions are provided. The proposed hierarchy provides users with the ability to describe and query object motion and interactions at different levels. The performance of automatic segmentation and EMU retrieval based on representative affine motion parameters was demonstrated by examples. We also show that EMUs and ERUs can facilitate semantic-level video summarization using an interactive authoring tool.

References

[1] F. Idris and S. Panchanathan, "Review of image and video indexing techniques," *J. Vis. Commun. Image Represent.*, vol. 8, no. 2, pp. 146–166, 1997.

[2] R. Lienhart, W. Effelsberg, and R. Jain, "VisualGrep: A systematic method to compare and retrieve video sequences," *Proc. SPIE*, vol. 3312, pp. 271–282, 1998.

[3] M. M. Yeung, B. L. Yeo, W. Wolf, and B. Liu, "Video browsing using clustering and scene transitions on compressed sequences," *Proc. SPIE*, vol. 2417, pp. 399–413, 1995.

[4] B. L. Yeo and M. M. Yeung, "Classification, simplification and dynamic visualization of scene transition graphs for video browsing," *Proc. SPIE*, vol. 3312, pp. 60–70, 1998.

[5] P. Alshuth, T. Hermes, L. Voigt, and O. Herzog, "On video retrieval: Content analysis by ImageMiner," *Proc. SPIE*, vol. 3312, pp. 236–247, 1998.

[6] Y. Deng and B. S. Manjunath, "NeTra-V: Toward an object-based video representation," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 616–627, Sept. 1998. Special Issue on Segmentation, Description, and Retrieval of Video Content.

[7] S. F. Chang, W. Chen, H. J. Meng, H. Sundaram, and D. Zhong, "A fully automated content-based video search engine supporting spatiotemporal queries," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, pp. 602–615, Sept. 1998.

[8] A. M. Tekalp, P. van Beek, C. Toklu, and B. Gunsel, "Two dimensional mesh-based visual object representation for interactive synthetic/natural digital video," *Proc. IEEE*, vol. 86, no. 6, pp. 1029–1051, 1998.

[9] H. Gu, Y. Shirai, and M. Asada, "Motion description and segmentation of multiple moving objects in a long image sequence," *IEICE Trans. Inform. Syst.*, vol. E78-D, no. 3, pp. 277–289, Mar. 1995.

[10] R. Jain, R. Kasturi, and B. G. Schunck, *Machine Vision*. Englewood Cliffs, NJ: Prentice-Hall, 1995.

[11] C. Toklu, A. T. Erdem, M. I. Sezan, and A. M. Tekalp, "Tracking motion and intensity-variations using hierarchical 2-D mesh modeling for synthetic object transfiguration," *Graph. Models Image Process.*, vol. 58, no. 6, pp. 553–573, Nov. 1996.

[12] Y. Fu, A. T. Erdem, and A. M. Tekalp, "Tracking visible boundary of objects using occlusion adaptive motion snake," *IEEE Trans. Image Processing*, vol. 9, pp. 2051–2060, Dec. 2000.

[13] P. van Beek, A. M. Tekalp, N. Zhuang, I. Celasun, and M. Xia, "Hierarchical 2D mesh representation, tracking, and compression for object-based video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 353–369, Mar. 1999.

[14] A. M. Ferman and A. M. Tekalp, "Efficient filtering and clustering methods for temporal video segmentation and visual summarization," *J. Vis. Commun. Image Represent.*, vol. 9, no. 4, pp. 336–351, 1998.

[15] A. M. Tekalp, *Digital Video Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1995.

[16] L. S. Shapiro, *Affine Analysis of Image Sequences*. Cambridge, U.K.: Cambridge Univ. Press, 1995.

[17] J. Y. A. Wang and E. H. Adelson, "Representing images with layers," *IEEE Trans. Image Processing*, vol. 3, pp. 625–628, Sept. 1994.

[18] Y. Altunbasak, P. E. Eren, and A. M. Tekalp, "Region-based parametric motion segmentation using color information," *Graph. Models Image Processing*, vol. 60, no. 1, pp. 13–23, 1998.

[19] C. S. Regazzoni and A. Teschioni, "A new approach to vector median filtering based on space filling curves," *IEEE Trans. Image Processing*, vol. 6, pp. 1025–1037, July 1997.

[20] J. Astola, P. Haavisto, and Y. Neuvo, "Vector median filters," *Proc. IEEE*, vol. 78, no. 4, pp. 678–689, 1990.

[21] G. Sudhir and J. C. M. Lee, "Video annotation by motion interpretation using optical flow streams," *J. Vis. Commun. Image Represent.*, vol. 7, no. 4, pp. 354–368, 1996.

[22] J. Z. Li, M. T. Ozsu, and D. Szafron, "Modeling of video spatial relationships in an object database management system," in *IEEE Proc. Int. Workshop Mult. Data. Manag. Syst.*, 1996.

[23] M. Egenhofer and R. Franzosa, "Point-set topological spatial relations," *Int. J. Geograph. Inform. Syst.*, vol. 5, no. 2, pp. 161–174, 1991.

Yue Fu was born in Beijing, China, in 1967. He received the B.S. degree in electrical engineering from Tsinghua University, Beijing, in 1991, and the M.S. degree in electrical engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, in 1994.

From April 1994 to June 1996, he was with National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, working in the areas of computer vision, image processing, and pattern recognition. Beginning in September 1996, he was with the University of Rochester, Rochester, NY, pursuing the Ph.D. degree in electrical and computer engineering, working in the areas of digital video analysis and description. He passed away on July 6, 1999.

Ahmet Ekin (S'01) was born in Antalya, Turkey, in 1979. He received the B.S. degree with highest honors in electrical engineering from Bogazici University, Istanbul, Turkey, in 1999. He received the M.S. degree in March, 2001 from the University of Rochester, Rochester, NY, where he is pursuing the Ph.D. degree in electrical and computer engineering.

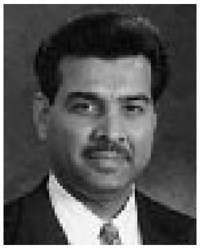
His research interests are object-based video representations, real time object tracking, and semantic event detection. He has been a Consultant for the Eastman Kodak Company, Rochester, since September 1999. During Summer 2001, he was a summer intern at AT&T Labs, Middletown, NJ.

Mr. Ekin is a Student Member of the IEEE Signal Processing and Computer Societies.

A. Murat Tekalp (S'80–M'82–SM'91) received M.S. and Ph.D. degrees in electrical, computer, and systems engineering from Rensselaer Polytechnic Institute (RPI), Troy, NY, in 1982 and 1984, respectively.

He was a Senior Research Scientist with the Eastman Kodak Company, Rochester, NY, from 1984 to 1987. He joined the Electrical Engineering Department at the University of Rochester in September 1987, where he is currently a Distinguished Professor. Since June 2001, he has held a Professor position at Koc University, Istanbul, Turkey. His current research interests are in the area of digital image and video processing, including object-based video representations, image/video segmentation, motion tracking, image/video restoration and super-resolution, image/video compression, multimedia content description, digital watermarking, and secure media. He served as an Associate Editor for the Kluwer journal *Multidimensional Systems and Signal Processing* (1994–2000); Area Editor for the *Journal of Graphical Models and Image Processing* (1995–1998). He was on the editorial board of the *Journal of Visual Communication and Image Representation* (1995–2000). He is the Editor-in-Chief of the *EURASIP Image Communication Journal*. He is the author of *Digital Video Processing* (Englewood Cliffs, NJ: Prentice-Hall book 1995) and holds five U.S. patents.

Dr. Tekalp received the NSF Research Initiation Award in 1988 and was named as Distinguished Lecturer by the IEEE Signal Processing Society in 1998. He chaired the IEEE Technical Committee on Image and Multidimensional Signal Processing (1996–1998). He is a founding member of IEEE Technical Committee on Multimedia Signal Processing. He is a member of the IEEE Signal Processing Society Conference Board. He was appointed Special Sessions Chair for the 1995 IEEE International Conference on Image Processing (ICIP), the Technical Program Co-Chair for IEEE ICASSP 2000 held in Istanbul, and General Chair for IEEE ICIP 2002 to be held at Rochester, New York. He is the founder and first Chairman of the Rochester Chapter of the IEEE Signal Processing Society. He was elected as the Chair of the Rochester Section of IEEE in 1994–1995. He has served as an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING (1990–1992) and the IEEE TRANSACTIONS ON IMAGE PROCESSING (1994–1996).



Rajiv Mehrotra is the Program Manager for the Media Asset Management Program of the Entertainment Imaging Division of Kodak and the Technology Manager for Media Asset Management R&D Program of Kodak R&D Labs, Rochester, NY. Prior to joining Kodak, he held faculty positions at the University of South Florida, Tampa, the University of Kentucky, Lexington, and the University of Missouri, Columbia. He is co-editor of the book *The Handbook of Multimedia Information Management,* (Englewood Cliffs, NJ: Prentice-Hall, 1997).

Dr. Mehrotra was co-editor of a special issue of the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING on multimedia information systems (August 1993). He also co-edited a special issue on image database management for *IEEE Computer*(December 1989). He is on the editorial boards of the IEEE MULTIMEDIA and *Pattern Recognition Journal* and has served on the program/organizing committee of several international conferences.