Design and Implementation of a Video Database System Supporting Semantic Contents-Based Scene Retrieval

Taketoshi USHIAMA
Kyushu Institute of Design
4-9-1 Shiobaru, Minami-ku. Fukuoka, 815-8540 Japan

Jyunji SUZUKI and Toyohide WATANABE
Nagoya University
Furo-cho, Chikusa-ku, Nagoya, 464-8603 Japan

Abstract

This paper describes the design and implementation of a video database system that demonstrates our research on video data modelling, and semantic content-based video retrieval.

Video is the medium that is suitable for recording dynamic aspect of the real world and is used in various domains: entertainment, education, and so on. Recent progress of computer technology enables users to manipulate videos on personal computers easily, and video database systems have been expected to manage large quantities of video efficiently and effectively. One of the most important technologies in video database systems is the content-based retrieval because it provides high usability of the system.

Contents of video can be classified in two categories: *visual content* and *semantic content*. Up to now, the many successful techniques based on visual contents (colour, shape, texture, movement, and so on) for video data modelling, indexing, and retrieving video segments are proposed. On the other hand, relatively few techniques have been proposed based on semantic contents. In this paper, we focus on the meaningful temporal video segment (called *scene*), and introduce a video database system for retrieving various kinds of scenes based on semantic contents. We concentrated on the video database of baseball games as an example of possible application

The traditional approach for the video data modelling is the *interval-based indexing*, in which keywords are assigned to the meaningful intervals in video. This approach has a drawback that end-users hardly retrieve scenes on the basis of their viewpoints because previously the retrievable scenes have been decided statically. In order to overcome this drawback, we introduce *Event-Activity Model*, which allows end-users to compose scenes dynamically according to their interests. This model is designed on the bases of the *frame-based indexing* approach, in which indexes are assigned to the frame.

The most important modelling element of Event-Activity Model is *event*, which represents characteristic transitions of properties on entities in videos. For example, “a pitcher throws a ball” is an event in baseball games. The semantic contents of a video are represented as an event sequence called *context*. The meaningful subsequences of context are called *activity*. For example, let us assume that a context on a baseball game contains the following subsequence: 1) “a pitcher throws a ball”, 2) “a batter hits a ball”, 3) “a ball passes over a fence”; this subsequence represents the activity “home run”. Figure 1 depicts this example. If events are assigned to a frame, the scene can correspond to an activity. Therefore scene retrievals can be processed by means of subsequence matching on context, and queries of users can be represented as *automata*. This model supports three different viewpoints of

users: 1) the difference of entities which users focus on, 2) the difference of granularity of composite activities, and 3) the difference of abstract levels in which activities are captured.

Our video database system based on Event-Activity Model consists of the three repositories: the video repository, the index repository, and the context repository. The video repository stores digitised videos and provide facilities for composing scenes according to the scene specification, which is described as a pair of start and end frames. The context repository stores content descriptions of videos in the video repository and provides facilities for retrieving activities that the specified automata accept on context. The index repository maintains relationships between an event in context and a frame in video, and supports to convert a subsequence to a scene specification. The process for the scenes retrievals is as follows:

1. A user specifies his/her requirements as automata.
2. The context repository retrieves activities that the specified automata accept.
3. The index repository converts retrieved activities to scene specifications.
4. The video repository returns scene to user according to the specifications.

We implement a system prototype on SGI workstation O2 with the object-oriented programming language C++. This system supports some kinds of video stream including Motion-JPEG, MPEG-1, QuickTime and so on, and provides graphical user interfaces. The indexing window supports basic video operations (play, stop, fast-forward etc.) and the composition and assignment of events by menu selection. The video query window enables users to specify and to conform their queries as the forms of state transition graphs.

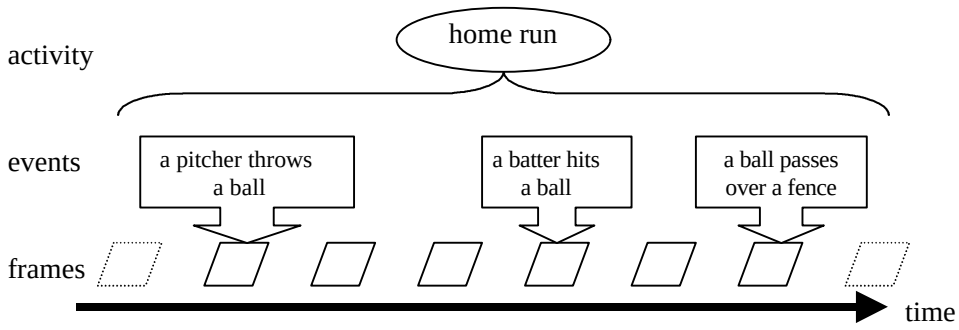

Figure 1: event and activiy