使用視訊序列比對的運動動態分析系統
Sport Motion Analysis System using Video Sequence Matching

張厥煒
Chueh-Wei Chang
靜宜大學資訊管理系
Department of Computer Science and Information Management, Providence University, Taichung, Shalu Taiwan, R.O.C.
cwchang@csim.pu.edu.tw

李素瑛
Suh-Yin Lee
交通大學資訊工程系
Department of Computer Science and Information Engineering, National Chiao Tung University, Hsinchu, Taiwan, R.O.C.
Sylee@csie.nctu.edu.tw

摘要

在此論文中，我們將呈現出視訊資訊系統用於運動動態分析中的數種重要分析方法．我們提出了在兩個視訊序列中，比較物件運動差異的多重子序列比對方法．經過切齊的處理後，我們可以找到兩視訊動態分析的最佳配對位置並計算出評估報告．
關鍵字：圖案比對，動態分析，視訊資訊系統

Abstract

In this paper, we present several important methods in a video information system for sport motion analysis. We propose a multiple subsequences matching mechanism for comparing the difference of object motion between two video sequences. After an alignment process, we can find the mapping of the best matching, and get an evaluation report between the two video sequences for motion analysis.
Keywords: Pattern Matching, Motion Analysis, Video Information System

1. Introduction

With rapid advances in data storage, image processing, telecommunication, and data compression, video has become an important component in the new generation of information system. Video data contains a large amount of spatial and temporal information. The information related to the position, distance, temporal and spatial relationship are included in the video data implicitly. Especially, the changes of video objects in equally divided temporal interval are quite useful for motion analysis and cannot be provided by other media easily. Therefore, a fast, easy use and cost effective video information system becomes an important task for the application of motion analysis. Furthermore, how to design a fast video matching and indexing mechanism is another important issue in video information systems.

In this paper, we present the methodology of designing a video information system for sport motion analysis. In Section 2, we give a brief introduction for the video information system and the sport motion analysis. In section 3, we propose several matching models for comparing the difference of object motion between two video sequences. In Section 4, we provide a sophisticate video content segmentation and indexing mechanism. In Section 5, we show our video information prototype system and sport motion analysis subsystem along with several query examples. In Section 6, we conclude the concluding remarks.

2. Video based Sport Motion Analysis System

A general *video information system* [1] is the information system that manages the video input, video processing, video query, video storage, and video indexing to provide a collection of video data for easy access by a large number of users. We have designed a video information system for multiple purpose use [4-6] before we enhance and apply this system to the domain of sport motion analysis. This video information system has the capability to analyze and retrieve video sequences based on the video contents [13].

Sport research has increased dramatically over the past few years [9]. Recent research has made tremendous contribution in improving a player's conditioning program, developing material strategies, designing better equipment, and understanding performance techniques. We believe that imitation is one of the best ways to get improvement in one's learning stage. Correct body posture is extremely important for sport performance. By using correct posture, a player can perform skills more efficiently and with less chance of injury. By comparing the posture frame-by-frame, a player can realize where he goes wrong and what the best choice is. If we can utilize the spatio-temporal characteristics of image sequences, we can have the capabilities of analyzing the activities of movement.

In most of the motion analysis experiments, researchers use a tripod-mounted high speed film camera that can take film shots at 100 frames per second to record the motion of the object as it moves across the field of view, and/or use a large number of sensors and wires to capture the slight movement of target objects. Furthermore, these experimental equipments are very expensive and/or restrict the performance of target objects. Therefore, we attempt to set up a new sport motion analysis environment with the help of digital video, a personal computer and a video information system.

Very few researches have been done in the sport motion analysis by computer based video. The VideoGraph [3] provided a good means for physics instruction, but it did not provide a multiple view ports in its user interface and neither a matching model for temporal data analysis. Therefore, designing a video information system that can analyze the motion phenomena, figuring out the variation, and collecting a large amount of data for searching and prediction, are important steps toward success.

3. Content Alignment and Matching

3.1. Measurement of Distance and Matching Model

After the feature extraction process [2,5], we can compare the target video with a standard sample video and evaluate the difference between them. First, we need to define the measurement of the frame distance and the sequence distance between two video sequences for a specific feature. If two video sequences are similar in feature values and are of same duration, the sequence distance will be the summation of individual frame distance between corresponding frames. For the target sequence $A = a_1a_2...a_n$ and the sample sequence $B = b_1b_2...b_m$, where n and m are the length of sequences A and B, respectively. If $m = n$, the sequence distance can be simply obtained by

$$\sum_{i=1}^{m} |a_i - b_i| \qquad (1)$$

We can also get a **mapping function** $F : [m] \rightarrow [n]$. If $m = n$, it will be a one-to-one and onto mapping function $F(i) = i$.

In practice, it is hard to shoot video just right on the starting position of a series of movement even with a postprocess of editing. Furthermore, even if we can shoot video in a fixed rate, e.g. 30 frames per second, we still do not know where the starting frame of each target movement is in a video sequence. A point can be matched to any point in a sequence having the same value. This is the one-dimensional aperture problem in motion analysis community. Therefore, the position correspondence between two video sequences is a crucial task to be solved.

The comparison between two video sequences is not just simply a fixed length template matching as the shift matching model. For each kind of video sequence matching model, two things need to be solved before calculating the sequence distance: the video frame alignment point and the video frame correspondence. That is, if $m < n$ and $q-p+1 \neq m$ occur, where should we start the distance measurement, where is the left and right alignment point, p and q, respectively, and how can we get the mapping function, what is the mapping function $F : [(1,2,...,m)] \rightarrow [(p,p+1,...,q)]$?

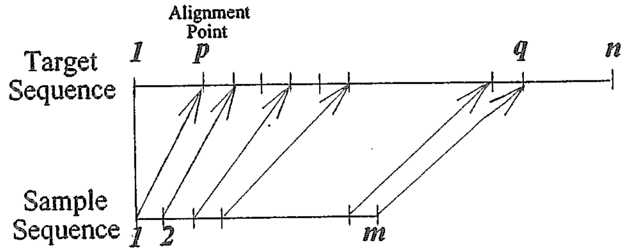


Figure 1. Complicate matching model

3.2. Complicate Content Correspondence

A more realistic and complicate situation is that we need to skip some of the video frames, as shown in Figure 1, then to find the best matched position and the best mapping. This situation is caused by the bad timing of movement (too fast or too slow). Unfortunately, the approximate pattern matching algorithm does not work when the case $q-p+1 > m$ occurs.

Therefore, we use the algorithm in the optimal correspondence of string subsequence (OCS) [12] to solve our sequence correspondence problem. We slightly change the definition in OCS algorithm to meet our requirements. The sequence distance of two sequences A' and B' is now redefined as :

$$S_{F'}(A', B') = (m'+n'-2r)\tau + \sum_{\forall i, F'(i) \neq \perp} |a'_i - b'_{F'(i)}| \qquad (2)$$

where m' and n' are the length of sequences A' and B', respectively; r is the number of matched pairs and τ is the error tolerance. The best matched distance between two sequences is defined as:

$$D(A', B') = \min_{\forall F'} S_{F'}(A', B') \qquad (3)$$

The new definition of mapping function is a mapping such that $F(i) = \perp$ or $(F(i) \neq \perp$ and $F(j) \neq \perp$ and $i < j) \rightarrow (F(i) < F(j))$. $\perp$ stands for the mismatch in a position. The minimum distance mapping can be expressed as follows. Given two sequences A' and B', find a mapping function F of A' and B' such that the sequence penalty is minimized. In order to find out the required mapping function F of A' and B', we build a minimum penalty table which accumulates the information of correspondence, and from the table we can trace back the correspondence and construct the desired F. The penalty table construction algorithm is depicted in Algorithm 1. The mapping function construction algorithm is shown in Algorithm 2.

An example of penalty table for sample sequence {268, 261, 173, 128, 73} from frame 18 to frame 22 and target sequence {265, 258, 163, 118, 115, 76} from frame 23 to frame 28 is shown in Table 1, where τ is 40. The mapping function is F : {(18, 23), (19, 24), (20, 25), (21, 26), ($\perp$, 27), (22, 28)}. The frame 27 in target video is skipped in this mapping and the sequence distance is 69.

```
Algorithm 1. Penalty Table Construction
Input. An target sequence target[1..n], a sample sequence sample[1..m],
and the similarity threshold τ.
Output. A penalty table penalty[1..n, 1..m].
Method.
penalty[0, 0] = 0;
for i = 1 to n do penalty[i, 0] = i* τ;
for j = 1 to m do penalty[0, m] = j* τ;
for i = 1 to n do
  for j = 1 to m do
     penalty[i, j] = min{ penalty[i-1, j-1] + abs(target[i] - sample[j]),
                          penalty[i-1, j] + τ, penalty[i, j-1] + τ };
return penalty[n, m];
End-of-Algorithm Penalty Table Construction.
```

```
Algorithm 2. Mapping Function Construction
Input. A penalty table penalty[1..n, 1..m], and the similarity threshold τ.
Output. A mapping function set mapping[].
Method.
i = n; j = m;
while (i <> 0 and j <> 0) do
 begin
  if penalty[i, j] = penalty[i-1, j] + τ  then i = i - 1;
  else
   if penalty[i, j] = penalty[i, j-1] + τ  then j = j - 1;
   else
    begin
     add (i, j) to mapping[];
     i = i - 1;  j = j - 1;
    end;
 end;
return mapping[];
End-of-Algorithm Mapping Function Construction.
```

Table 1. Penalty table of group S-7 and Q-6

Value	Tar	(23)265	(24)258	(25)163	(26)118	(27)115	(28)76
Sample	0	40	80	120	160	200	240
(18) 268	40	***3***	43	83	123	163	203
(19) 261	80	43	***6***	46	86	126	166
(20) 173	120	83	46	***16***	56	166	206
(21) 128	160	123	86	56	***26***	***66***	86
(22) 73	200	163	126	96	66	68	***69***

3.2. *Multiple Subsequences Matching*

When the frame numbers of two video sequences become large, the calculation of penalty table will be time consuming. In order to reduce the computation time, we attempt to segment video sequences into multiple subsequence groups by multiple alignment points. Therefore, the matching model for multiple subsequences as shown in Figure 2 is more sophisticate than the complicate matching model. After presenting the definitions of sequence distances and solving the case of complicate matching model, we still need an efficient mechanism to overcome this the most common and useful case - the multiple subsequence alignment and matching problem.

If we can specify the corresponding subsequence groups, the overall distance can be defined as follows:

$$M(A,B) = \sum^{g} D'(A^i, B^i) \qquad (4)$$

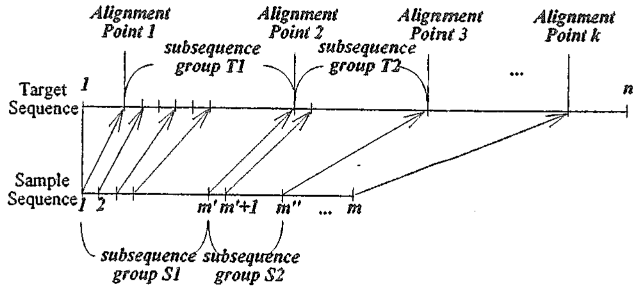


Figure 2. Matching model for multiple subsequences

where g is the number of groups; A^i and B^i are the ith subsequences of target sequence A and sample sequence B, respectively.

4. Content Segmentation and Indexing

4.1. *Video Content Segmentation*

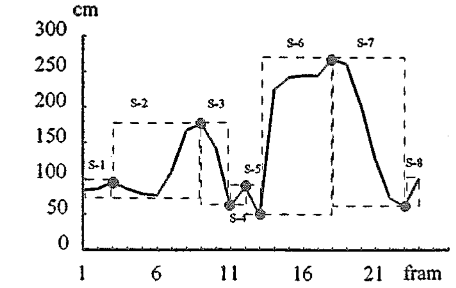
The best way to find the alignment points is to segment the curve according to the prominent feature points in the sequences. In [6], we proposed a segmentation mechanism by using curve features in video sequences. Several kinds of curve features can be found according to the changing of a sequence. These points with changing features are good positions for the alignment points. Basically, we classify curve features into four categories. They are: strongly up edge in a sequence, strongly down edge in a sequence, increase out of range, and decrease out of range. We use the segmented prominent points as the alignment points. After the segmentation process, a sequence can be divided into several subsequences enclosed by bounding boxes.

We use the most similar bounding boxes between target and sample sequences as the major subsequence alignment point, and propagate the mapping of alignment points box-by-box into left and right side of the sequence. For example, the Q-6 and S-7 in Figure 3 are the major subsequence alignment points of target and sample sequences, respectively.

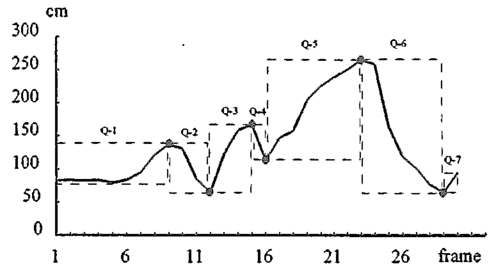
4.2. *Time-series Video Indexing*

The content-based indexing of time-series sequence is a nontrivial and challenging problem [8,13]. To speed up the specific feature point searching and subsequence alignment processing, we use B-tree [7] as our index structure [6]. We can search and find a possible major subsequence alignment point from the bounding box segmentation points with the help of this video indexing. We can use the prominent point as the key of index structure. The best choice of threshold value is dependent on the distribution of feature values and the density of prominent points.

Also, if we assume that two successive video frames in the same sequence are quite similar, the major portions of the video sequence remain unchanged, and the time intervals between frames are short enough that many



(a) One sample sequence



(b) Target sequence

Figure 3. Frame/value diagram with bounding box and prominent feature points

changes do not occur at once. In particular, this means that the feature value of an object in the video sequence changes gradually. Several special characteristics exist: (1) Two frames with a large difference tends to indicate that a special motion has occurred. In other words, an extremely large distance change means a special event happened. (2) A special event can also be defined as the feature value in a single video frame in a specific predefined range.

5. System Implementation and Query Issues

5.1. Sport Motion Analysis Subsystem Prototype

We have implemented a video information prototype system supporting sport motion analysis as shown in Figure 4 and Figure 5 on an IBM-PC compatible computer with MS-WINDOWS 95 by using C++ language to test the object extraction and approximate matching mechanisms.

The video information prototype system provides the capabilities of random access, frame stilling, frame stepping, and slow play for an interactive video system [14]. The Playback Area has the functions of mark in/out logging, preview window, video file playback, and displaying of current processing frame. The Thumbnail Area can act as a display of six-divided frames, or a set of still salient images of active video sequences, or an A-to-F roll editing monitor each with a different video file, or the video icons of query results. The Image Processing Area shows the results of video/image processing functions (for example, color key, caption, special effects, etc.), temporarily duplicated still frame, and region-based color feature extraction. The Single Frame Data Area shows the feature values in a current processing frame (for example, histogram, region size, etc.). The Curve Data Area shows the evaluation curve, bounding boxes, query and matched curve. The Video Parameter Area shows the related parameters of a current active video file. The Annotation Area shows the default annotation about video contents. The Status Bar Area shows the processing percentage, current cursor coordinates and corresponding R, G, B color intensity, feature value of video sequences, and video editing mark in/out cue points. The Menu Bar Area provides the functions of annotation, feature extraction, indexing, query processing and database management.

The sport motion analysis subsystem provides multiple view ports, measurement tools, measurement results display and matching mechanisms. Users can easily check the motion differences file-by-file and frame-by-frame.

5.2. Query Issues

We go back to the volleyball court, then start our queries. The spike in volleyball is one of the most difficult sport skills to perform. The most important element of good execution is proper timing. Three common errors in executing an attack are (1) beginning the approach too early, (2) lacking height on the jump during spike, and (3) contacting the ball behind the hitting shoulder. If an attacker approaches the ball too early, two possible results are: (1) to stop, wait for the set, and thus lose the benefit of the approach; or (2) to back up to attack the ball because he has approached too far.

Therefore, the major purpose of our sport motion analysis will be how to take a good spike from a given location on the court. According to the result of the best mapping, we can calculate the posture and timing differences, and generate a report about the mismatched posture and significant differences between the target and sample sequences. That is, we can observe that, in the sample sequence, the preference hand swings very fast and with large motion but, in the target sequence, the attacker seems to act slower than in the sample sequence. This result is concluded by the differences of the stable frames (frame 9-22 in the sample sequence and frame 9-28 in the target sequence), by a large overall distance 447 (the summation of sequence distance: 76, 83, 40, 179, and 69, for respective stable subsequence groups), and by the minimum/maximum range for each alignment group. Furthermore, we can also ask the questions, such as:

(1) What is one's highest spike position? We can answer this question by automatic measuring of the maximum altitude of preference hand.

(2) What is one's jumping offset in this spike? -- By measuring the X axis offset when jumping altitude is nonzero.

(3) How long can one stay in the air? -- By measuring the number of frame when jumping altitude is nonzero.

jumping offset divided by jumping duration.

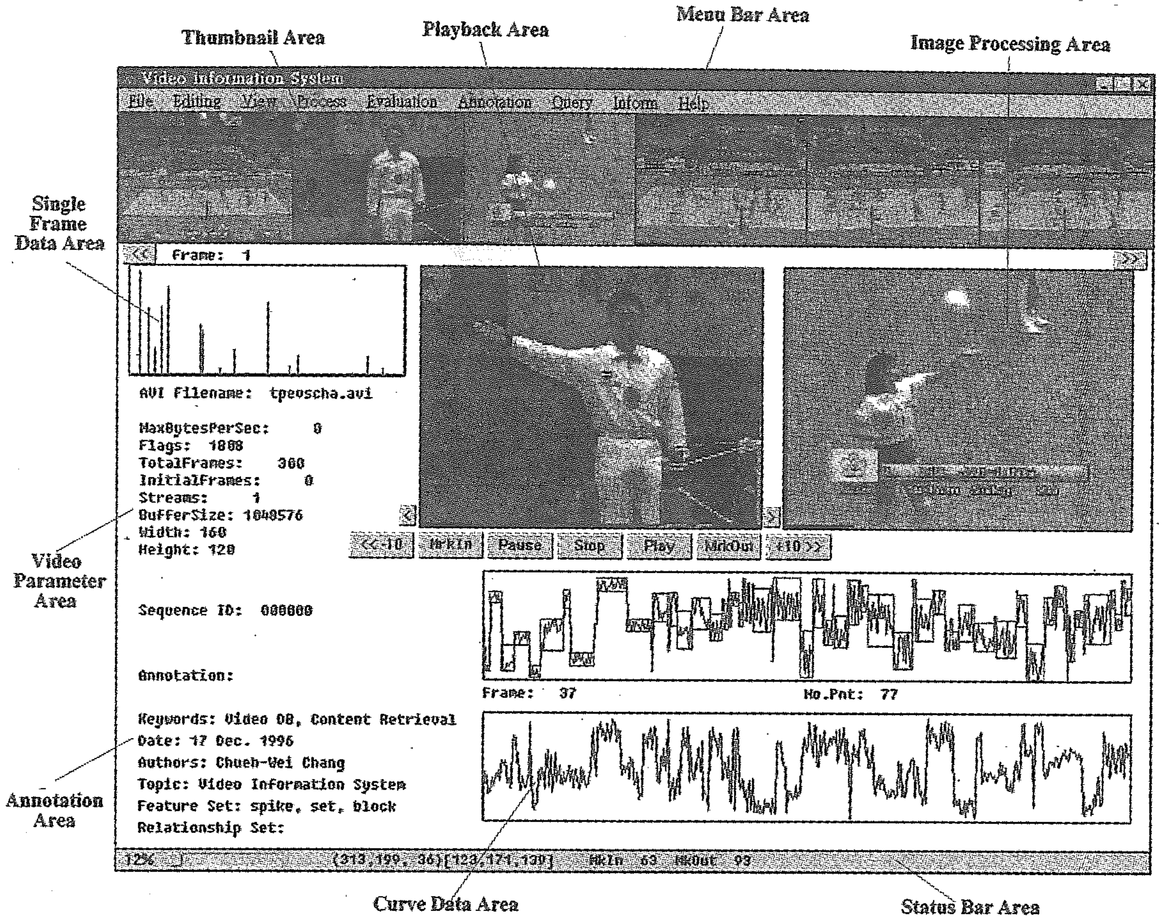


Figure 4. Video information system prototype

(4) What is one's average forward velocity when jumping in the air? -- By measuring the

(5) What is the minimum leg angle when one starts to jump? -- By measuring the leg position and calculating the leg angle manually.

Some of the answers can be obtained automatically from the system, such as (1)-(4) and some of them can only be obtained manually by inspecting the video sequences frame-by-frame.

6. Conclusions

In this paper, we attempt to set up a sport motion analysis environment with the help of digital video and personal computer, and try to maximize performance and enjoyment of the sport in the possibly shortest time. By providing content-based retrieval, applications of digital video are broad in many aspects. Video records the changes of scenes according to time. The changes of video objects is quite useful for dynamic scene and motion analysis. Change of objects between different frames provides much information about the behavior of these objects in the video.

Video object segmentation, time-series data matching and indexing are the essential preliminary steps in most video information systems. The methods discussed in this paper are representative of techniques commonly used in practice. We have successfully analyzed several sport motion examples using the developed system. In addition, the techniques used for object feature extraction, curve segmentation and matching appear general in nature. Thus we are hopeful that suitable modifications will support the system to analyze more complex video, such as color video sequences of simplistic outdoor sport scenes.

The spatio-temporal matching algorithm can be extended to several useful application domains, such as gesture recognition [10]. We only need to record the spatio-temporal feature values, then the recognition processing can be replaced by a searching method with an error tolerance. This approach will be faster and more flexible than traditional ways. We leave this gesture recognition project as our future work.

References

[1] S. Adah, K. S. Candan, S. H. Chen, K. Erol and V. S. Subrahmanian (1996) The advanced video information system: data structures and query processing. *Multimedia Systems* 4, 172-186.

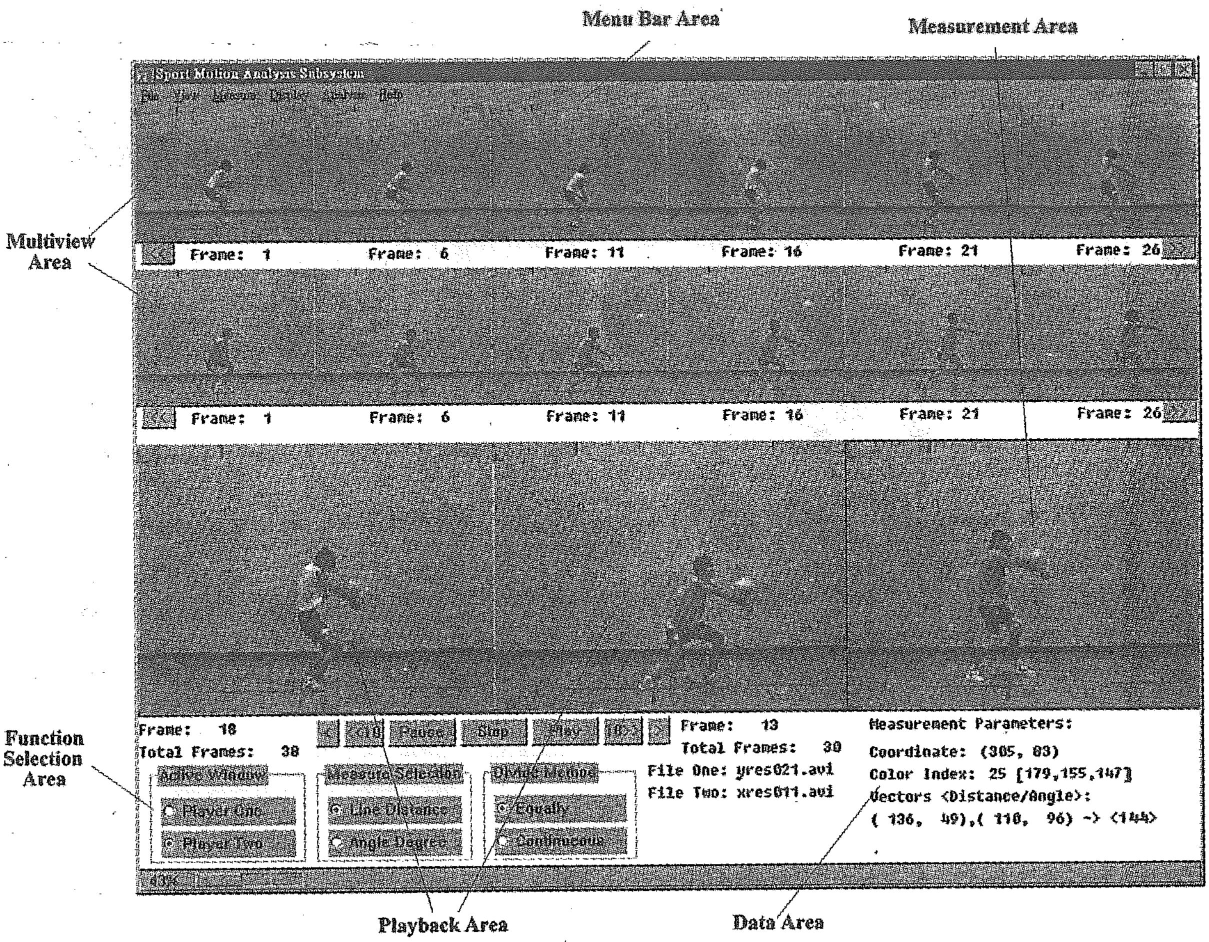


Figure 5. Sport motion analysis subsystem

[2] B. Bascle, P. Bouthemy, R. Deriche and F. Meyer (1994) Tracking complex primitives in an image sequence. In: *IEEE 12th Intl. Conf. on Pattern Recognition.* Jerusalem, Israel, pp. 426-431.

[3] R. Beichner, M. DeMarco, D. Ettestad and E. Gleason (1989) VideoGraph: a new way to study kinematics. In: *Computers in Physics Instruction* (E. Redish and J. Risley, eds) Addison-Wisley, Raleigh, NC, pp. 244-245.

[4] C. W. Chang, K. F. Lin and S. Y. Lee (1995) The characteristics of digital video and considerations of designing video databases. In: *Fourth Intl. Conf. on Information and Knowledge Management.* Baltimore, Maryland, pp. 370-377.

[5] C. W. Chang and S. Y. Lee (1995) Statistical and topological feature extraction and matching in video sequences. In: *National Computer Symposium.* Chung-Li, Taiwan, pp.693-700.

[6] C. W. Chang and S. Y. Lee (1996) Indexing and approximate matching for content-based time-series data in video database. In: *First Intl. Conference on Visual Information Systems.* Melbourne, Australia, pp. 567-576.

[7] D. Comer (1979) The ubiquitous B-tree. *ACM Computing Surveys* 11, 121-137.

[8] C. Faloutsos, M. Ranganathan and Y. Manolopoulos (1994) Fast subsequence matching in time-series databases. In: *ACM SIGMOD.* Minneapolis, MM, pp. 419-429.

[9] C. Frohlich (1986) Resource letter PS-1: physics of sports. *Am. J. Phys.* 54, 590-593.

[10] P. Kelly and S. Moezzi (1995) Project reports: visual computing laboratory. *IEEE Multimedia* Spring, 94-99.

[11] D. Koller, J. Weber, T. Huang and S. Russel (1994) Towards robust automatic traffic scene analysis in real-time. In: *IEEE 12th Intl. Conf. on Pattern Recognition.* Jerusalem, Israel, pp. 126-131.

[12] Y. P. Wang and T. Pavlidis (1990) Optimal correspondence of string subsequences. *IEEE Trans. on Pattern Analysis Machine Intelligent* 12, 1080-1087.

[13] J. K. Wu, A. D. Narasimhalu, B. M. Mehtre, C. P. Lam and Y. J. Gao (1995) CORE: a content-based retrieval engine for multimedia information systems. *Multimedia Systems* 3, 25-41.

[14] D. Zollman and R. Fuller (1994) Teaching and learning physics with interactive video. *Phys. Today* 47, 270-274.