A Low-Cost Real-Time Tracker of Live Sport Events

Janez Perš[1], Goran Vučkovič[2], Stanislav Kovačič[1], Branko Dežman[2]

[1]Faculty of Electrical Engineering, Tržaška 25
{janez.pers},{stanislav.kovacic}@fe.uni-lj.si
[2]Faculty of Sport, Gortanova 22
{goran.vuckovic},{brane.dezman}@sp.uni-lj.si
University of Ljubljana, SI-1000 Ljubljana, Slovenija

Abstract

We present a low-cost prototype system for real-time tracking of players during a sport event. The system was tested on recordings of a squash singles match. Two players are visible from bird's eye perspective as they move around the squash court. System recovers position of each of the players on each frame and presents the results in real time on computer screen. Method for automatic tracker intialization is proposed. Our setup makes processing speed of up to ten frames per second possible, using low cost, widely available PC hardware.

1. Introduction

The human motion analysis is receiving increasing attention from computer vision researchers [1]. Most sports involve complex human motion and therefore automated capturing, analyzing and quantifying the ability of the athlete can offer significant help to the sports expert [2]. On the other hand, gathering of various information about the sport activity *during* the action can help the audiences to better appreciate the sport, if this information is presented to the audience in real-time.

However, real-time acquisition of important data, such as athlete position, differs substantially from acquisition for the purpose of later analysis by sport experts and researchers. Reliability and possibility of uninterrupted operation are crucial in this case, and there is usually no time or possibility to correct tracking errors by human supervisor.

Therefore, most successful commercial solutions are not computer vision based, for example the TRAKUS system [3], which is based on the array of microwave receivers, which analyze the signal, emitted from special transmitters. As players need to wear special devices, this technique is not allowed in many sports, which are governed by rigid and conservative rules.

Real-time computer vision based tracking system for use in sport has to fulfill the following requirements:

- It should use computationally inexpensive algorithms.
- It should be general enough to work in different environments without significant parameter tweaking.
- It should be able to start and initialize itself with minimum available knowledge about the observed scene.

These requirements are hard to fulfill, especially when the nature of most sports is considered: they feature multiple athletes moving over a large area. Therefore, implementation of such system for use in team sports (for example footbal and handball) is difficult; some degree of operator intervention is needed as many of the players on the court simply *look* the same on images, captured with standard video equipment [2].

We have chosen squash for the purpose of prototyping the real-time player tracker system. During the squash match, two players are always present on the court. Unlike tennis, both players may move around the whole court, and from the computer vision point of view, collisions occur.

This paper is structured as follows: first, we present the problem of tracking players in game of squash. Next, we propose image subtraction based method that solves our problem and yields satisfactory performance. As this method relies on some parameters, which are usually provided by supervisor, we propose automatic methods for obtaining those parameters as well. In the last part of the paper we present the developed prototype tracking system, which integrates data acquisition with results presentation, in the form that might be interesting for sport spectators. Finally, some conclusions about the possibility of implementing the commercial real-time tracker application are given.

2. Acquired images

Squash [4] is singles or doubles game played in a four-walled court with a long-handled strung racket and a small rubber ball. Squash is usually played by two people, but it can be played by four (doubles). In this paper, we focus on the case of singles version of squash.

The requirements for the squash court are extremely strict [5]. The court measures 9750×6400 milimeters. The clearance between the court plane and the ceiling should be no less than 5640 milimeters. The whole court should be well and uniformly illuminated, in such way that no shadows are cast. Several markings are present on the court plane, and their dimensions are well defined, which makes them perfectly suitable for camera calibration.

Uniform lightning requirements simplify the image processing task, as shadows are distracting elements in the case of indoor machine-vision based athlete tracking [2]. Required height of the ceiling reduces the required field of view, and therefore reduces the barrel distortion of acquired images. Example image from the actual squash match is shown in Fig. 1. Images were acquired in color.

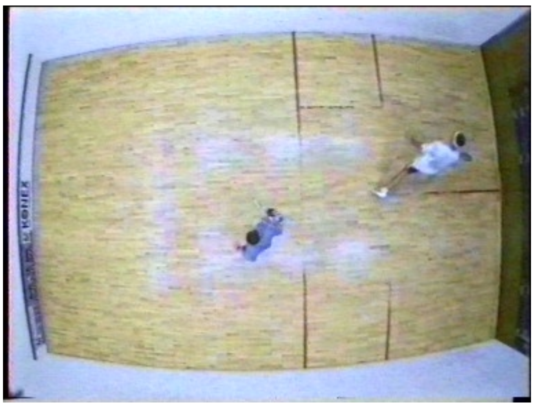

Figure 1. Image, obtained during the squash match. Camera was mounted in the hole in the ceiling.

Players occupy relatively large part of the image, and their structure is visible. However, to track the motion of their extremities, different camera angles would be necessary. In our case, we modelled the player as simple blob.

3. Tracking

Proposed tracking method includes several steps. Let us assume that the image of the empty court is a priori available to the tracker. The tracking method can be then outlined as follows:

- Calculate the difference between the background (court) RGB image and the current RGB image, using the following formula:

$$\begin{aligned} D(x,y,t) = & \ |I_r(x,y,t) - R_r(x,y)| + \\ + & \ |I_g(x,y,t) - R_g(x,y)| + \\ + & \ |I_b(x,y,t) - R_b(x,y)|, \end{aligned} \qquad (1)$$

$$(x = 1..M,\ y = 1..N)$$

  where I_r, I_g and I_b denote red, green and blue component of current image, respectively, and R_r, R_g and R_b red, green and blue component of the background image. $M = 384$ and $N = 288$ denote the dimensions of images used.

- Threshold the image using the threshold value T. The threshold value can be input manually at the beginning of the tracking process, or calculated during the tracking process, as described in the next section.

- Use the 3×3 median filter on the thresholded image in several iterations to reduce noise. Players will appear as blobs after this step.

- Delete the points that are equally distant from both players. Player position from previous image is used for this purpose. In case of player collision this procedure divides one bigger blob, which belongs to both players. The result can be seen in Fig. 2.

- Sort the blobs using the area they occupy as a key. The gravity centers of two largest blobs represent positions of each of the players.

Proposed method is capable of tracking players without any hints about player position. In case that players are close to each other on the first frame (and blob separation technique does not work), the tracking starts as soon as players separate.

Figure 2. Blob separation. Black line represents pixels that have equal distance to previous positions of both players.

The output of this method are raw positions of the players in the image coordinate system. As the camera is stationary, relations between court coordinate system and image coordinate system can be obtained in advance using the court marks. Different methods can be used for that purpose. The method we used is described in details in [2].

4. Tracker initialization

The proposed algorithm has a serious drawback: both the background image and the threshold value T have to be specified before the processing of each image takes place.

Background image (image of the empty court) can be obtained manually in advance, when no players are present on the court. However, this is impractical. Another possibility would be a continous background recovery [6]. However, much simpler technique can be used for that purpose. At the start of tracking, a sequence of several (10 or more) images are captured with longer intervals as the actual framerate used for tracking. The median operator along the temporal dimension is used on each channel of each pixel on the image. If players were moving during the acquisition of the image sequence, the result will be an empty playing court.

The threshold value T can be obtained for each image using a priori knowledge of the scene. Different threshold values result in different number of positive pixels. If the positive pixel count is higher than estimated area that both players should occupy on the scene, the threshold T is increased for the certain amount. If the count is lower, value of T is decreased for the same amount. The estimated area of the players depends on the image geometry and because of the well defined dimensions of the court we expect that it could be set to a constant level in advance of the start of the tracking process.

5. Prototype tracker implementation

The prototype tracking system was implemented using a Pentium II 400 MHz equipped personal computer, running Windows NT. Computer was equipped with Motion-JPEG video acquisition interface, which was used to digitize 120 second video sequence to a M-JPEG stream.

Additionally, computer was equipped with inexpensive PCI 3D graphic accelerator device (3Dfx Voodoo Graphics based board). Tracker software was written in Borland Delphi, using Intel Image Processing Library for the most of the basic image processing tasks.

3D graphic accelerator provided us with the VGA output for the second VGA monitor, which was used to display results of a tracking in a visually attractive manner, using the features of the 3D accelerator. Primary system monitor was used to control the tracking software operation.

During the tests, the software timer was set to read the digitized Motion-JPEG sequence frame by frame, decoding the JPEG data and processing the image. This way, we simulated the real time tracking process, in which the video acquisition hardware takes the role of a processing trigger. Another timer was used to trigger 3D rendering of the output image on the secondary computer monitor. The output image consisted of source image of the squash court, with overlaid results of the tracking. In the lower part of the image, a 3D rendered scene of a court, with both players represented as a semi-transparent blocks was rendered. The sample output image is shown in Fig. 3.

6. Results

The entire 120 second sequence was processed using the described setup. Background image was prepared in advance, using the method described. Every second frame from the sequence was used, which would correspond to the processing rate of 12 frames per second. Processing timer interval was set to 100 miliseconds, which resulted in full load of the main processor. Processing time per frame averaged 100 miliseconds. 30

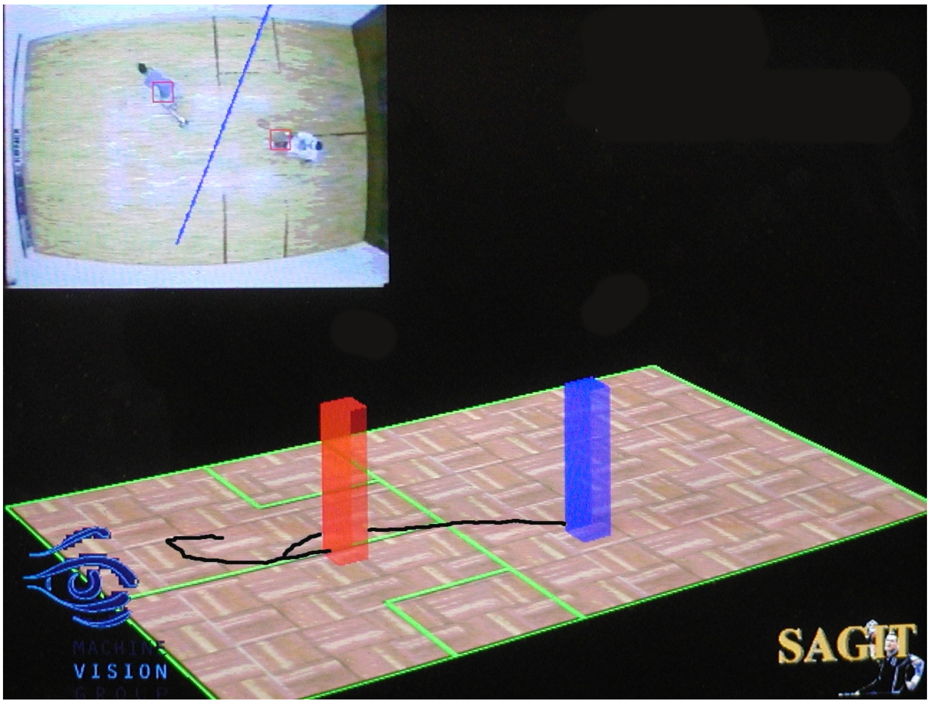

Figure 3. Sample output image on the secondary monitor.

miliseconds of this time were spent on Motion-JPEG frame decoding, which would not be necessary if a RGB framegrabber would be used. This time could be reduced by compressing the Motion-JPEG stream with lower bitrate setting. 3D rendering and display accounted for just 3% of the total processor load, as the main burden of triangle rendering was shifted to the 3D accelerator.

7. Conclusion and future work

We have shown that real time sport event tracking could be performed. Many requirements exist for such a system, most notably the need for automatic initialization. Tracking problem can be successfully solved for some of the sports, other will require fundamentally different approaches.

Future work will include total automation of initialization setup by implementing background image recovery in the software itself. Additionally, since we provided the mechanism to decode Motion-JPEG frames on the fly, the adaptation to live video processor using inexpensive consumer grade video acquisition boards is trivial. Another intriguing opportunity would be a use of 3D rendering device with a video-out port. It would enable our sytem to act as a 'black box', with video of an actual match as an input and 3D rendered video as an output.

References

[1] J. K. Aggarwal and Q. Cai. Human motion analysis: A review. In *IEEE Nonrigid and Articulated Motion Workshop*, pages 90–102, Puerto Rico, June 17-19 1997.

[2] J. Perš and S. Kovačič. Computer vision system for tracking players in sports games. In *Proceedings of the First Int'l Workshop on Image and Signal Processing and Analysis IWISPA 2000*, pages 81–86, Pula, Croatia, June 14-15 2000.

[3] Trakus inc. home page. Internet URL. `http://www.trakus.com`.

[4] World Squash Federation. *The Rules of World Singles Squash*. `http://www.worldsquash.org/rules.html`, April 2001.

[5] World Squash Federation. *Squash specifications for courts, racquets, balls*. `http://www.worldsquash.org/rules.html`, June 1999.

[6] I. Haritaoglu, D. Harwood, and L. S. Davis. Fast background scene modelling and maintenance for outdoor surveillance. In *Proceedings of the 15^{th} International Conference on Pattern Recognition - ICPR 2000*, volume 4, pages 179–183, Barcelona, Spain, September 3-8 2000.