PREPROCESSING OF BALL GAME VIDEO SEQUENCES FOR ROBUST TRANSMISSION OVER MOBILE NETWORK

Olivia Nemethova*, Martin Zahumensky**, Markus Rupp*

*TU Wien, Institute of Communications and RF Engineering
Gusshausstr. 25/389, 1040 Wien Austria
{onemeth, mrupp}@nt.tuwien.ac.at
**TU Bratislava, Institute for Telecommunications
Ilkovicova 3, 812 19 Bratislava Slovakia
zahumensky@ktl.elf.stuba.sk

ABSTRACT

Ball games like soccer belong to most popular TV programs. Due to the lossy nature of the wireless channel and high compression rates necessary to match the given bandwidth it is difficult to transmit such programs over mobile networks in real-time. As the important information is carried by a single small object - the ball - it is necessary to ensure its correct reconstruction at the receiver. In this article we propose a robust sender-side preprocessing method, based on the ball recognition and tracking, enabling correct rendering of the ball at the receiver.

1 INTRODUCTION

Emerging 3rd generation of mobile communication systems brought new multimedia services. One of the most interesting applications is video-streaming, already provided by many operators all over the world. Here, sport programs are of particular interest, whether as part of news or as stand-alone and possibly live broadcast transmissions. Beyond doubt, the most popular sport programs are the ball games such as soccer. However, the transmission of live streaming ball game video sequences over mobile network introduces several challenges. Spatial and temporal smoothness of video sequences allows for high compression performed at the sender before the transmission. This compression results in a certain quality degradation. Streaming services are delay sensitive and therefore they are usually transported via the unreliable User Datagram Protocol (UDP) rather than via the Transmission Control Protocol (TCP), the latter providing the possibility of transport layer retransmissions. UDP usage leads to possible packet losses at the receiver, further degrading the end-user quality. To match the screen of common mobile terminals, a resolution QCIF (144×176) is used. For PDAs and laptops the CIF (288×352) resolution is of relevance and considered in this investigation as well. The most important object in a ball game is understandably the ball. Ball games are usually recorded using a slightly moving wide-angle camera. This leads to situations, in which the ball is represented by three or four pixels only and thus very susceptible to any kind of degradation, which has also considerable effect on the user perceptual quality [1]. Some snapshots of our ball game video-sequences are shown in Figure 1.

Figure 1: Snapshots of some sequences 'soccer1', 'soccer2', 'ice-hockey' and 'squash' with different ball types.

In case of video-streaming over wireless networks, the receiver typically is power and size limited mobile terminal. Therefore it is not feasible to implement complex postprocessing methods, allowing to cope with the given problem.

The intention of this article is to provide a novel, simple and robust method to cope with this problem. In Section 2 possible methods of preprocessing are discussed and compared. Section 3 presents the simulation setup to test some of them. In Section 4 the results are presented and interpreted. Section 5 contains conclusions and some final remarks as well as an outlook in the future.

2 BALL TRACKING METHODS

There are more possibilities how to protect the ball and how to ensure its display at the right place at the re-

ceiver. However, to treat the ball separately, automatic recognition and tracking of the ball is required.
Such task introduces several challenges because:

- The ball game video-sequence usually contains cuts or slow-motion replay parts.
- The ball is small, especially for the relevant QCIF and CIF resolutions.
- There can be more than one object resembling a ball.
- The ball does not have to appear in every frame: it can be covered by the players or there can be parts of video without it (audience, details on players).
- The appearance of the ball changes over the time (zooming, shadow, occlusion).

Some of the ball appearances during the video sequence can be seen in Figure 2.

Figure 2: Snapshots of different balls: three soccer balls, two squash balls and one ice-hockey ball.

Previous work has been performed on soccer video analysis; a helpful state-of-the-art description and proposal for automatic soccer video analysis can be found in [4]. The purpose of that work was an event detection (goal, penalty or red/yellow card detection), one can find there efficient methods how to detect the scene change or slow-motion replay, shot classification is mainly based on the detection of the players and playground lines but not on the ball recognition.

Another simple method for event detection in a soccer game is proposed in [5]. The detection is based on the tracking of the trajectory of the candidate objects. The most consistent trajectories and corresponding objects are then detected as a ball. Several known object recognition methods were used to detect the ball. For instance in [7] the recognition based on the circle detection or in [6] the component analysis was used.

The purpose of the above mentioned methods was first of all event detection. Our aim is to protect the smallest and most important object the ball. We need to avoid wrong detection. The most critical situation occurs in the frames where the ball is visible within the playground not surrounded by any other objects. After using a high compression, the ball often seems to disappear by blurring fluently into the grass. To overcome this situation a correct ball detection is required. Since real-time is further required, the method needs to be simple with low complexity. The circular shape of the ball can be used for the decision but it is not possible to only rely on this single information. Especially in case of a wide-angle camera, parts of the sequence and QCIF resolution the ball consists often of not more than 3-4 pixels.

2.1 SAD based detection

The simplest way to detect the ball is to use the sum of absolute differences (SAD) [2] as a measure of similarity between the model ball and the searched region:

$$\mathrm{SAD}(n,x,y) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} |F_n(x+i, y+j) - T(i,j)|, \quad (1)$$

where F_n is the n-th frame of the video sequence; x, y are the SAD coordinates (within the searched region of the frame), i, j are the coordinates within the $N \times M$ template (model ball) T. The search algorithm can then sequentially count the SAD within the region of interest and select the minimum as a candidate position of the ball. With this simple method, problems occur if the ball size changes (e.g. by zooming) or if the ball is partially overlapped. As will be shown later, this can be partly overcame for higher resolutions by another approach presented in [3] (applied there to the face recognition).

2.2 Detection based on chromatic histograms

The original method in [3] works within the HSI color space with two chromatic components (Hue and Saturation) and one gray-scale component (Intensity). The relation between RGB and HSI color-space is as follows:

$$I = \frac{1}{3}(R+G+B), \quad (2)$$

$$S = 1 - \frac{3}{R+G+B}[\min(R,G,B)], \quad (3)$$

$$H = \cos^{-1}\left\{\frac{\frac{1}{2}[(R-G)+(R-B)]}{[(R-G)^2+(R-B)(G-B)]^{1/2}}\right\}. \quad (4)$$

In general, for chromatic histograms, any color space might be used, that has one gray-scale and two chromatic components (e.g. YUV).

Following steps are to obtain the location of the ball:

1. Compute the histograms M_i for a ball model and I_i for the i-th frame.
2. Compute the histogram ratio $R_i = \min[M_i/I_i, 1]$.
3. Transform the input image R_i into the backprojected image B_i: $B_i(x,y) = R_i(I(x,y))$, where $I(x,y)$ represents the color value at (x,y) of the input image.
4. After blurring the backprojected picture, the location with peak value corresponds to the most likely location of the model object within the image.

3 PROPOSED SYSTEM

First, we implemented a method based on the SAD similarity measure. Input in our algorithm is the video sequence and one or more ball models. As an initial ball we took the ball that first appeared in the sequence together with a ball boundary containing a small part of the playground. The block diagram of the procedure is shown in Figure 3.

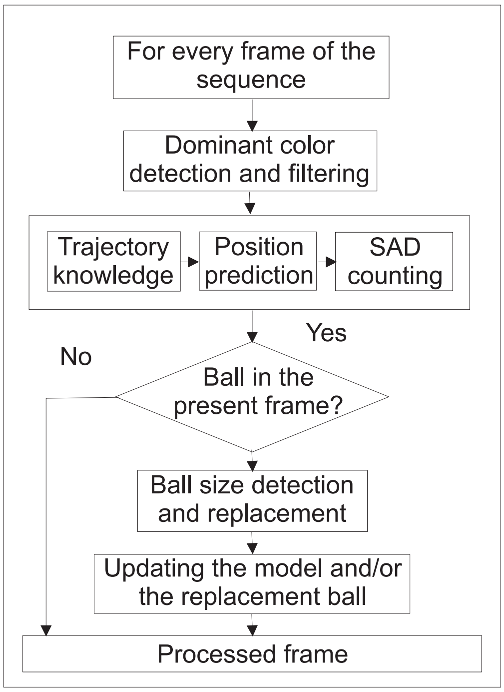

Figure 3: Block diagram of the actions to be performed upon every frame.

We performed dominant color detection and filtering to ease the ball detection. First, the dominant color is detected (playground) and then the whole playground is set to this color and a low-pass filter is applied to avoid the possible segments of playground lines or players. Performing the detection, one can further spare the complexity and enhance the robustness by tracking the ball coordinates over time and simple linearly predicting the ball location in the next frame. The search algorithm begins then in the predicted location and continues following a coil as can be seen in Figure 4. We used the trajectory knowledge as an input in the decision about the ball location.

After recognizing the ball, it is of advantage to update the ball model as it can change during the video sequence for instance by different illuminance of the playground. Update may be done simply by taking the previous ball, or preferably by weighted averaging over past balls. Also the ball replacement can be updated according to the size/color of the detected ball. We do not

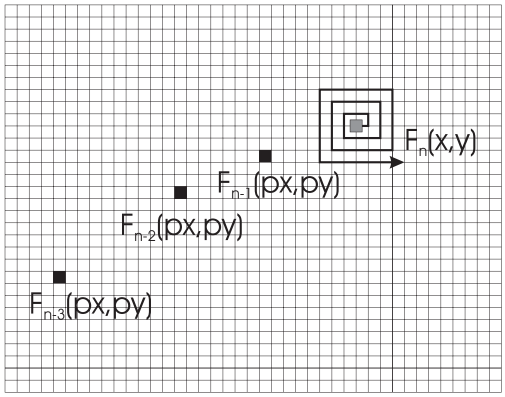

Figure 4: Counting of SAD along the coil for every pixel of the searched region in n-th frame $F_n(x, y)$ with the starting point linearly predicted from the previously detected ball locations $F_{n-i}(px, py)$.

need to consider the detail pictures where the ball is big enough not to be degraded by encoding. Therefore, the SAD based method works properly, too.

We also implemented the method based on chromatic histograms. This method we only tested on CIF resolution sequences because in QCIF sequences the ball is too small to allow reliable color analysis. For this method we also used the information about the ball shape and trajectory to avoid wrong decisions.

For both methods we also set the threshold to decide whether the ball appears in the picture at all to avoid wrong detections. After detection we replaced the ball with an appropriate larger model. The size of the replacement ball was chosen with respect to the amount of compression to be used for encoding, too (larger ball for higher compression).

4 RESULTS EVALUATION

For testing our methods we used diverse video sequences of the following ball games: soccer, ice hockey and squash. Some snapshots of original, compressed and preprocessed soccer sequences are depicted in Figure 5. As an encoder for these sequences we used the free Nokia Multimedia Player [9] encoding the $7s$ short sequences as an MMS of 96kB size.

A snapshot from an ice hockey sequence compressed with and without preprocessing is shown in Figure 6. Please note, that in the case of the compressed ice hockey without preprocessing the puck is not visible at all. It is even difficult to recognize it in the original QCIF resolution video sequence. The method based on chromatic histograms we tested only on the soccer sequence. In the case of ice hockey the playground is usually covered by advertisements that make it difficult to track the ball using this method. Squash sequences are very challenging for the ball tracking in general as the ball is very small and moves fast. The rapid movement causes the ball sometimes appearing as a short line instead of a point. As the ball in our sequences was black and so can be the parts of the background or players, it was difficult to handle it by the method based on chromatic histograms. The usage of a ball with an expressive color might help the situation (at some matches neon-yellow balls are used). The camera capturing squash is usually static, causing the ball disappearing from the picture and reappearing from an almost arbitrary side of the court. This reduces the efficiency of the ball tracking essentially.

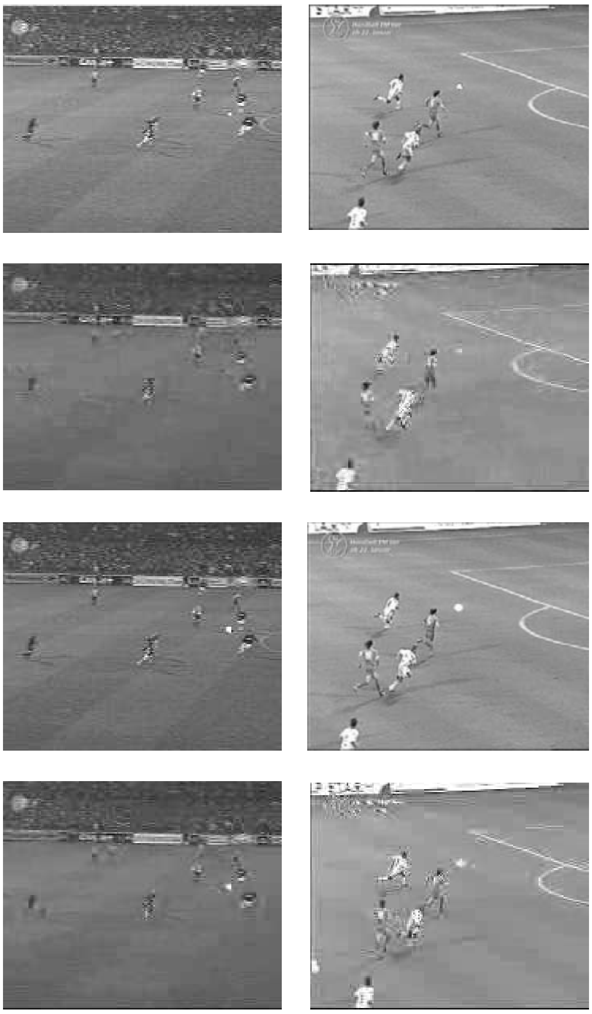

Figure 5: The effect of preprocessing. Snapshots from two (left and right) soccer sequences. From the top: original frame, compressed frame, original preprocessed frame, frame compressed after preprocessing.

4.1 Objective Evaluation

The following table presents the performance of the proposed detection method based on SAD only.

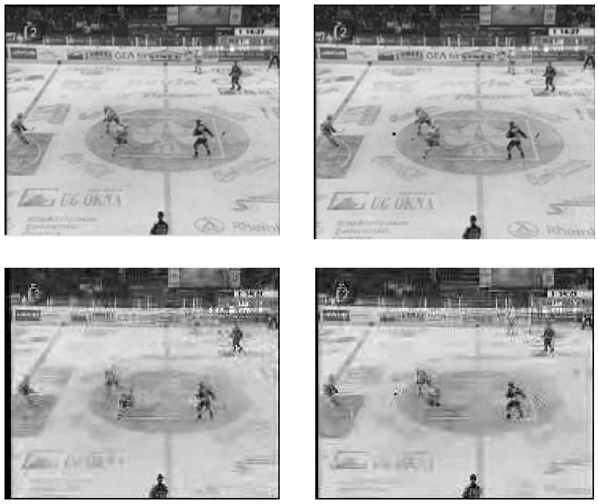

Figure 6: The effect of preprocessing. Snapshots from an ice hockey sequence. Top: original and preprocessed frame without compression; Bottom: compressed original and compressed preprocessed frame.

Type	**Size**	**Correct**	**Wrong**	**Undet.**
Soccer	QCIF	96.23%	0.91%	2.86%
Soccer	CIF	97.24%	0.67%	2.09%
Ice-Hockey	CIF	93.31%	3.78%	2.91%
Squash	CIF	85.14%	2.38%	12.48%

The values are counted over all used test sequences.

In the following table, the performance of the proposed detection method based purely on chromatic histograms can be seen.

Type	**Size**	**Correct**	**Wrong**	**Undet.**
Soccer	QCIF	94.79%	1.02%	4.19%
Soccer	CIF	98.02%	0.55%	1.43%

The results for QCIF soccer sequences were slightly better, using the SAD based method due to the small ball size. The method based on chromatic histograms gives better results with CIF resolution but the differences are small and the SAD based method also gives sufficient results preserving the low complexity. Most of the wrong detections and undetected balls occur if the ball is partially covered by a player. In future, this could be improved by occlusion detection (by tracking the position of the players). Nevertheless, the SAD method gives very good results even if compared for example with [7] working with higher resolution sequences.

For the ice hockey sequence there are more wrong detections. It is caused mainly by the fact, that the end of the hockey stick resembles the puck in some of the pictures. If the puck is near to it, the trajectory tracking algorithm cannot correctly distinguish them.

4.2 Subjective Evaluation

To obtain the user subjective perceptual quality evaluation by means of mean opinion score (MOS), we performed short tests with 10 unpaid voluntary test persons. Tests were done according to [8]. We performed absolute category rating tests for short (ten second duration) soccer video sequences with QCIF resolution encoded by an H.264 encoder, using different bit rates (28, 32, 56 and 64kbps) and a constant frame rate 10fps. We displayed the sequences on the screen of a Sony-Ericsson Universal Mobile Telecommunications System (UMTS) mobile terminal, depicted in Figure 7.

Figure 7: Sony-Ericsson UMTS mobile terminal Z1010 used for subjective perceptual quality tests.

The test subjects were asked to evaluate the presented videos on a five-grade scale (1-excellent, 2-good, 3-medium, 4-poor, 5-bad). The subjects were not informed that some of the videos are preprocessed, the video sequences were presented in an arbitrary order. We did not show the original to the test subjects. The results for MOS averaged over all test persons are presented in the following table:

Bit Rate [kbps]	With prep.	Withot prep.
28	4.182	4.909
32	3.455	4.636
56	2.636	3.909
64	1.818	3.545

It can be seen, that the visibility of the ball has an essential influence on the user perceptual quality evaluation. With preprocessing we obtained better results for 32kbps sequences than for the 64kbps sequences without preprocessing. During the tests we presented also a sequence with the ball replaced by a ball that was unnaturally enlarged. The results confirmed, that in such case the user evaluation was worse than in the original compressed sequences where the ball could not be seen at all in some pictures. The best visual results we achieved for 56 and 64kbps by just sharpening the colors of the ball. For 32 and 28kbps also slight enlargement was appropriate.

5 CONCLUSIONS

Our aim was to ensure that despite of the high compression necessary for transmission over mobile networks, the user will be able to watch the ball games and see the small ball clearly even if wide-angle camera was used. In our work we focused on a **simple** and **robust** ball detection algorithm. If the ball is correctly recognized, we enlarge and sharpen it, so that it is visible well after the lossy compression and the transmission over the channel. The advantage of such method is that the preprocessing is only necessary at the sender side. Although we used a very simple, low-complexity approach for the ball detection, we achieved an essential improvement, visible from the presented subjective metric.

References

[1] O. Nemethova, M. Ries, E. Siffel, M. Rupp, "Quality Assessment for H.264 Coded Low-Rate and Low-Resolution Video Sequences", accepted to IASTED Internat. Conf. on Communications, Internet and Inf. Technology (CIIT) 2004.

[2] K.N. Ngan, C.W. Yap, K.T. Tan, "Video Coding for Wireless Communication Systems" Marcel Dekker Inc., Basel, 2001.

[3] T.W. Yoo, I.S. Oh, "A fast algorithm for tracking human faces based on chromatic histograms," Elsevier Pattern Recognition Letters, vol. 20, pp. 967-978, 1999.

[4] A. Ekin, A.M. Tekalp, R. Mehrotra, "Automatic Soccer Video Analysis and Summarization," IEEE Transactions on Image Processing, vol. 12, no. 7, pp. 796-807, July 2003.

[5] X. Yu, C. Xu, H.W. Leong, Q. Tian, Q. Tang, K.W. Wah, "Trajectory-Based Ball Detection and Tracking with Applications to Semantic Analysis of Broadcast Soccer Video," Proc. of ACM Multimedia Conference, Berkeley, USA, Nov. 2-8, 2003.

[6] M. Leo, T.D. D'Orazzio, A. Distante, "Independent Component Analysis for Ball recognition in Soccer Images," Proc. of the IASTED International Conference on Intelligent Systems & Control, Salzburg, Austria, pp. 351-355, June 25-27, 2003.

[7] T.D. D'Orazzio, M. Leo, M. Nitti, G. Cicirelli, "A real time ball recognition system for sequences of soccer images", Proc. of the IASTED International Conference On Signal Processing, Pattern Recognition, and Applications, Crete, Greece, pp. 207-212, 25-28 June, 2002.

[8] ITU-T Recommendation P.910, "Subjective video quality assessment methods for multimedia applications," Sept. 1999.

[9] Nokia Multimedia Player, available online at Nokia webpages: http://www.nokia.com/.