MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Collaborative Tracking of Objects in EPTZ Cameras

Faisal Bashir and Fatih Porikli

TR2006-088 March 2007

Abstract

This paper addresses the issue of multi-source collaborative object tracking in high-definition (HD) video sequences. Specifically, we propose a new joint tracking paradigm for the multiple stream electronic pan-tilt-zoom (EPTZ) cameras. These cameras are capable of transmitting a low resolution thumbnail (LRT) image of the whole field of view as well as a high-resolution cropped (HRC) image for the target region. We exploit this functionality to perform joint tracking in both low resolution image of the whole field of view as well as high resolution image of the moving target. Our system detects objects of interest in the LRT image by background subtraction and tracks them using iterative coupled refinement in both LRT and HRC images. We compared the performance of our joint tracking system with that of tracking only in the HD mode. The results of our experiments show improved performance in terms of higher frame rates and better localization.

SPIE, Video Coding & Image Processing (VCIP), 2007

Collaborative Tracking of Objects in EPTZ Cameras

Faisal Bashir and Fatih Porikli*

Mitsubishi Electric Research Laboratories, Cambridge, MA, USA

ABSTRACT

This paper addresses the issue of multi-source collaborative object tracking in high-definition (HD) video sequences. Specifically, we propose a new joint tracking paradigm for the multiple stream electronic pan-tilt-zoom (EPTZ) cameras. These cameras are capable of transmitting a low resolution thumbnail (LRT) image of the whole field of view as well as a high-resolution cropped (HRC) image for the target region. We exploit this functionality to perform joint tracking in both low resolution image of the whole field of view as well as high resolution image of the moving target. Our system detects objects of interest in the LRT image by background subtraction and tracks them using iterative coupled refinement in both LRT and HRC images. We compared the performance of our joint tracking system with that of tracking only in the HD mode. The results of our experiments show improved performance in terms of higher frame rates and better localization.

Keywords: Collaborative tracking, EPTZ tracking, Mean-shift analysis

1. INTRODUCTION

Automatic object detection and tracking from video sequences plays an important role in modern vision-based systems. The applications of this technique are immense including automatic video surveillance[6], aerial imagery[2], sports analysis[1], activity analysis[3], etc. The ability to spatially locate targets of interest and track the moving targets over a period of time is of central importance in these tasks. Most of the past and ongoing research in this area has been focused on tracking from single static or moving camera. Recently, the research efforts have focused more on the problem of collaborative tracking in the distributed environment[4]. One aim of the vision systems employing this approach is to cover a large field of view for tracking multiple targets at a fixed resolution by using multiple cameras. An added advantage is that tracking can be performed in severe occlusions under some homography constraints[5].

A lot of modern vision system applications are highlighting the importance of both wide coverage area as well as sharp detail on the moving target. One example of multi-sensor object detection and tracking is in video surveillance tasks. Here the goal is wide area monitoring on one hand, and acquiring high-quality biometric images on the other hand[6]. To identify people at a distance, a highly zoomed image is needed. But with high zoom, only a small portion of the area under surveillance can be monitored. To address this problem, Zhou et al[6] propose a master-slave architecture system. A static, wide FOV (master) camera is used to monitor wide area and detect moving humans. Upon detecting a human in FOV, the active narrow FOV pan-tilt (slave) camera is used to acquire high-resolution image of the human target and to perform tracking in narrow FOV. Their system is built using three standard PC systems for master camera processing, slave camera processing and pan-tilt unit control. On similar lines, Migdal et al[7] propose wide area high-resolution surveillance using a static wide FOV and an active narrow FOV PTZ camera unit. This focus-of-attention camera system is shown to cover a wide area for surveillance at high-enough resolution to perform moving object detection and tracking. It is shown in their presentation that for an equivalent level of high-resolution target tracking achieved by the stationary and PTZ camera system, a network of around 100 fixed FOV cameras will be required in one particular application setting studied by them.

Another major application area is sports video analysis. An example application is presented by Needham et al[1] for indoor soccer player tracking. In this domain, there is a growing interest in performing automatic play and player analysis. From the perspective of sports science industry, knowledge of players' movement patterns during play is an important benchmark for education and training. Sports broadcast industry is also interested in generating more dynamic content for end-users to more closely watch the movements and tactics of their favorite players. For most broadcast sports, the playing field is a fairly wide area which can not be covered by single camera systems with high-enough

* fatih@merl.com, Telephone: 1 617 621 7586

resolution. This observation is the motivation of our approach for collaborative tracking of sports players using EPTZ cameras. In this paper, we present a system for semi-automatic player detection and automatic tracking in the context of base-ball video. Our approach is domain-independent and can be applied to wide-area surveillance task; this point is illustrated by tracking results in outdoor surveillance application. Work presented in this paper is a step towards the ultimate goal of generating high resolution tracked imagery of players using single ETPZ camera.

This paper is organized as follows: section 2 lays out the problem domain and application scope including tracking and background modeling algorithms used in our implementation; section 3 presents our EPTZ solution with specific system outline; section 4 details the results of collaborative EPTZ tracking in our application domain; finally, conclusions and future work is presented in section 5.

2. PROBLEM STATEMENT AND BACKGROUND

This paper addresses the following problem: Given the HD video data of a play on a sport field (specifically base-ball in our case), perform player detection and tracking generating high-resolution imagery of the desired players as picked by the end-user. More specifically, we deal with a practical situation of the problem formulated herein: Due to massive amounts of data to be transferred, the HD camera and associated bandwidth is not capable of delivering HD quality images (1280x720) at full frame rate. Instead, at HD Mode the camera can deliver a low-resolution image of the whole FOV as well as a high-resolution image of the target region. We provide a solution that works under these constraints to deliver high-resolution tracked imagery of the semi-automatically detected players in the field.

Any system for player detection and tracking in sports videos has to deal with several challenging problems. The first problem is maintaining a wide area of coverage and a high-resolution image of the tracked player at the same time. These two requirements, as noted in the previous section, are at odds with each other. The second problem is automatically detecting the objects of interest (players) for further tracking. This issue is further complicated by the rather cluttered background in sports environments, owing mainly to the audiences outside the field. Apart from the cluttered background, another factor that contributes to further complicate the issue is the uncontrolled illumination changes and weather effects in the outdoor environments. Finally, once the objects of interest have been successfully detected spatially, an object tracking algorithm is needed to temporally locate the object in the video sequence frame by frame. This paper intends to address these three issues in the context of tracking moving objects in HD video sequences from an EPTZ camera.

2.1. Wide Area Coverage at High Resolution

The traditional solution for wide-area video surveillance is to set up enough number of narrow FOV static cameras in a collaborative network. With the constraint of obtaining high-resolution imagery of players (moving targets) in sports (video surveillance) application, this amounts to just scaling up the existing solutions to work on massive amounts of data. A solution based on simply scaling up the existing approaches is far from practical in most applications. In our application for baseball player tracking at HD1 resolution of 1280x720 pixels, around 16 HD cameras will be needed to tile the base-ball field the way it is covered by our system. This massive amount of data prohibits any existing solution due to the sheer volume. The requirements for data transmission and storage alone are prohibitive in modern applications. To address this problem, previous approaches have concentrated on developing static and PTZ cameras in master-slave architecture[6,7]. The disadvantage of that approach is that the two cameras have to be carefully calibrated to same world coordinate system. We address this issue in the context of modern cameras that support image outputs at multiple resolutions albeit with some time lag. We observe that recently, very high resolution cameras that are able to accommodate full frame rate video have become available in the vision research market. These cameras can provide more than 1Mega-pixels resolution and deliver exceptional details of the depicted scene. However, data transmission bandwidth and computational bottlenecks often limit the amount of video data to be analyzed at the user end for most existing systems. To accomplish the wide area of coverage at high-resolution, such cameras offer a mode of transmission that supports two video streams. One stream corresponds to a low-resolution thumbnail (LRT) of the over-all field of view, while the second stream delivers a high-resolution cropped (HRC) view of the target. In other words, the high-resolution cropped image acts as an electronic pan-tilt-zoom (EPTZ) camera. Further details of our solution are provided in the next section.

2.2. Object Detection using Background Modeling

The problem of automatically detecting the objects of interest for further tracking has been widely addressed in the recent literature. One approach for achieving this goal is using areas of motion in the scene to discriminate them from the background. Towards this goal, several successful approaches have been proposed that build a statistical representation of the background. A brief training period is required where statistics from a few frames are used to model the background appearance. Once the background model has been established, the incoming frames are compared with this model to mark the pixels belonging to moving objects. The background model should be robust to variations in background resulting from multiple time-varying natural phenomena. The variations in scene background arise from different sources, such as smooth and sudden illumination changes, windy (stagnant) conditions resulting in high (low) motion of natural objects like trees, waves in water, etc. To counter this problem in a robust manner, most of the existing approaches for background modeling rely on statistical representations. In this representation, the random process at each pixel from multiple frames is associated with a probability density function (*pdf*). The per-pixel *pdf* for background model can be represented parametrically using a specified statistical distribution that fits the data well. Alternatively, non-parametric approaches could be used for this representation. This stochastic representation of the random process at each pixel models the various appearances of the background effectively. On the lines of parametric background modeling, Stauffer and Grimson[8] proposed modeling the background with a mixture of Gaussians. The pixel-wise mixture of Gaussian approach models various forms of backgrounds effectively. Their background update method makes use of expectation maximization- (EM-) based framework for background learning. The background model update is performed at a pre-specified learning rate to dynamically adjust to changing conditions. Elgammal et al[9] argued to use non-parametric methods for density estimation to represent arbitrary distributions in a data-driven manner. They used kernel density estimation at each pixel to represent different background states.

Our background modeling technique is based on recursive Bayesian learning as proposed by Porikli et al[10]. In this approach, the background model is similar to Stauffer's pixel-based adaptive mixture model. The recent history of each pixel, $\{x_1, x_2, \cdots, x_t\}$, is modeled by a mixture of K Gaussian distributions. The probability of observing the current pixel value is:

$$P(x_t) = \sum_{k=1}^{K} \omega_{i,t} * \eta(x_t, \mu_{i,t}, \Sigma_{i,t}) \tag{1}$$

where K is the number of mixture components, $\omega_{i,t}$ is an estimate of the mixture weight of i^{th} Gaussian in mixture at time t, $\mu_{i,t}$ is the mean value of the i^{th} Gaussian in mixture at time t, $\Sigma_{i,t}$ is the covariance matrix of the i^{th} Gaussian in mixture at time t, and η represents the Gaussian probability density function:

$$\eta(x_t, \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(x_t - \mu_t)^T \Sigma^{-1} (x_t - \mu_t)} \tag{2}$$

A choice of $3 - 5$ for the parameter K is found to be sufficient for most application. For more dynamic scenes, more layers are required. In our formulation, instead of using EM for learning the parameters of the mixture, we use Bayesian recursive learning approach. Here, each pixel is defined as a layer of 3D multivariate Gaussians. In the RGB color space, each layer corresponds to a different appearance of the pixel. Using the Bayesian approach, we are not estimating the mean and variance of each layer, but the probability distribution of mean and variance. The background update algorithm maintains the multimodality (various appearances) of the background. At each update, at most one layer is updated with the current observation. After background model learning and update, foreground objects are detected by computing Mahalanobis distance of each pixel's observed color with confident background layers. Pixels that are outside of 99% confidence interval of all confident layers of the background are considered as foreground pixels. Finally, connected components labeling is performed on foreground pixels to mark the moving targets to be tracked.

2.3. Tracking of Detected Objects

The problem of reliably tracking the objects of interest becomes a lot simplified once foreground regions based on background models have been detected, and a change detection mask at each frame is generated. In the classical object tracking setting, a manually initialized object is to be tracked over time in a video sequence. Whereas, in our case, the results of object detection after background subtraction combined with user-input are used for object initialization. Given

two views of the scene (wide-area at low-resolution and narrow-area at higher-resolution), we highlight that joint tracking imposes certain unique requirements over the choice of tracking algorithms and which view to use. Depending on the original FOV size and amount of subsampling involved, the objects of interest could be very small to perform any meaningful tracking in the LRT view. On the other hand, because more data is available in the HRC view, more reliable tracking performance can be achieved in this view. Also, because of better resolution at the target, better object model update can be performed using data from HRC. Tracking in the HRC view has its own problems though. Since the HRC view gives a zoomed version of the object being tracked, its FOV is diminished. So, moving objects do not spend much time in the FOV spanned by HRC view. Also, since the object size to FOV size ratio is quite high in this view, a high-motion tolerant tracking algorithm is needed. As far as background generation and update is concerned, there seems to be little choice. Maintaining a high-resolution background model at the HD resolution is a time consuming task which becomes prohibitive in real-time system requirements. Due to these unique requirements, we use multi-kernel mean-shift tracking algorithm with foreground regions mask generated through background modeling in the HRC view. Also, the background image generated through LRT view is maintained using HRC view. In the next section, we tie the pieces together in the form of collaborative tracking system.

2.3.1. Multi-Kernel Mean-Shift Tracking with Foreground Mask

Mean-shift is a real-time algorithm for target tracking based on object appearance model. The tracking is based on a robust clustering technique which does not require prior knowledge of the number of clusters or their shapes. The algorithm starts on the data points and at each iteration, moves in the direction of maximum gradient. Iterations end when the point converges to a local mode of the distribution[12]. As pointed out earlier, the original mean-shift algorithm requires significant overlap on the target kernels in consecutive frames. This condition might not be met in the high motion areas of sports videos. To address this issue, we use the multi-kernel mean-shift algorithm[11] for tracking in the HRC view. In this approach, multiple kernels within a fixed radius of the original object location are initialized at high motion areas. Object template likelihood scores are computed at the converged points and the location associated with maximum score is marked as the object location.

Given the outline of detected object from change detection mask, the multi-kernel mean-shift first forms an object model for matching in successive frames. The object model is a nonparametric color template in the form of $W \times H \times D$ matrix whose elements are 3D color samples from the object. W and H are width and height of the object respectively and D is the size of the history window. Let Z_0 be the initial location of the object obtained through semi-automatic player initialization. If the object has been tracked up to current frame, it corresponds to the estimated location obtained through tracking from previous frame. We refer to the foreground pixels inside the estimated target box as $(x_i, u_i)_{i=1}^{N}$, where x_i is the 2D coordinate in the image coordinate system and u_i is the 3D color vector. Corresponding foreground sample points in the template are represented as $(y_j, v_{jk})_{j=1,k=1}^{M,D}$. Let $\{q_s\}_{s=1}^{m}$ be the kernel weighted color histogram of the template of initialized player to be tracked using multivariate normal kernel k_N for weighting. Let $p(z)$ be the color histogram of the candidate centered at location Z and let $b(z)$ be the background histogram at the same location. We construct background color histogram using only the confident layers of the background. The similarity between object model (template of player being tracked) and the candidate region is measured using modified Bhattacharya coefficient. This similarity measure includes background information:

$$\rho(z) = \alpha_f \sum_{s=1}^{m} \sqrt{q_s p_s(z)} - \alpha_b \sum_{s=1}^{m} \sqrt{b_s(z) p_s(z)} \tag{3}$$

where α_f and α_b are weights for foreground and background pixels. To locate the object in next frame, mean-shift vector at location Z_0 then becomes:

$$m(z_0) = \frac{\sum_{i=1}^{n} (x_i - z_0).w_i.g_N\left(\left\|\frac{x_i - z_0}{h}\right\|^2\right)}{\sum_{i=1}^{n} w_i.g_N\left(\left\|\frac{x_i - z_0}{h}\right\|^2\right)} \tag{4}$$

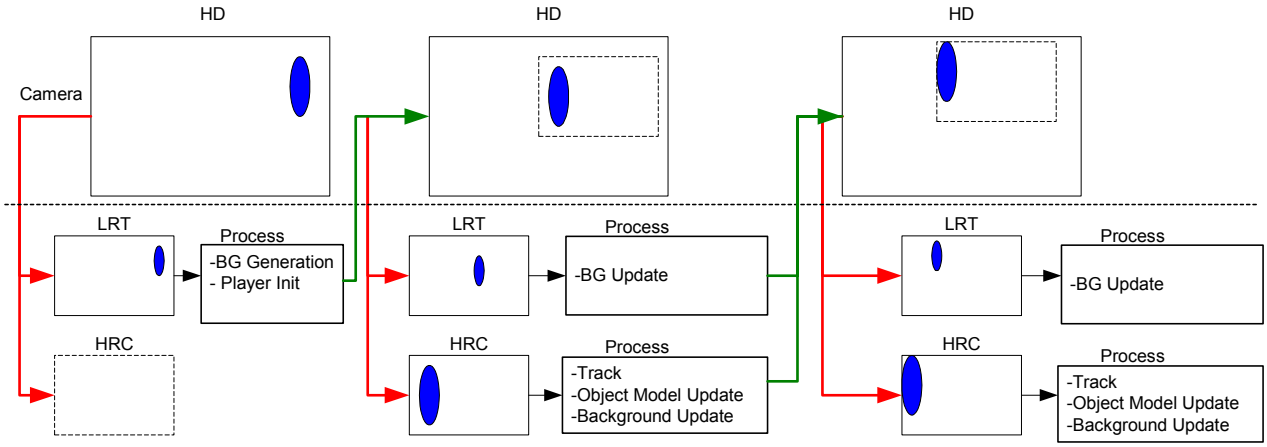


Figure 1. This figure shows the original HD image in the camera as well as the two images LRT and HRC transferred to system for processing.

where $g_N(x^*) = -k'_N(x^*)$, and w_i are the mean-shift weights derived through Bhattacharya similarity defined in Eq. (3) and h is the bandwidth of spatial kernel. Next, we compute the probability that a single pixel (x_i, u_i) inside the candidate target box centered at z belongs to the object. We compute this using Parzen window estimator on color distance between current target box and the object template history:

$$l(u_i) = \frac{1}{Dh_c^3} \sum_{k=1}^{D} k_N \left(\left\| \frac{u_i - v_{jk}}{h_c} \right\|^2 \right) \tag{5}$$

where h_c is the bandwidth of 3D color kernel, set to be 16 in our experiments. The final combined likelihood of an object being at location z is measured as:

$$L(z) = \frac{1}{N} \sum_{i=1}^{N} l_j(u_i) k_N \left(\left\| \frac{x_i - z}{h} \right\|^2 \right) \tag{6}$$

The kernel k_N assigns smaller weights to samples farther from the center of the object template making the estimation more robust. Object model update is handled in the HRC tracker since we have a lot more pixels to update the object model at high resolution. At the time of each update, the oldest samples of each pixel of the template (at D^{th} slice) are replaced with new ones. Based on foreground segmentation, template pixels corresponding to background pixels in current frame are not updated. Finally, scale adaptation of the objects is performed using the foreground pixels. Let B be the bounding box of the object centered at estimated location z_1. We define a second box O around the object center which has twice the area of B. The object scale is the solution of the maximization:

$$\begin{aligned} S &= \sum_{x \in B} \hat{c}(x) + \sum_{x \in O-B} (1 - \hat{c}(x)) \\ \hat{c}(x) &= \begin{cases} 1, & x \in \textit{foreground} \\ 0, & x \in \textit{background} \end{cases} \end{aligned} \tag{7}$$

The optimal bounding box of object is stored for object localization in the next frame.

3. ELECTRONIC PTZ SOLUTION

As briefly described in the previous section, the main challenge in our high-resolution player tracking is dealing with the images at two different resolutions effectively. Maintaining a robust background in the wide FOV using LRT image, we

insure wide-area coverage of our system. At the same time, tracking in the HRC image allows us to generate high resolution imagery of the tracked player for end-user display. Also, high resolution background image is maintained using information from LRT background and successive HRC images. An advantage of this multiple-resolution approach using the EPTZ camera is that the homography between low-resolution and high-resolution scene is trivially known. The high-resolution version of the target is merely from a scaled and cropped region of the low-resolution scene. Thus, unlike master-slave architecture of conventional mechanical PTZ camera tracking, no camera calibration step is required in the case of EPTZ camera-based system.

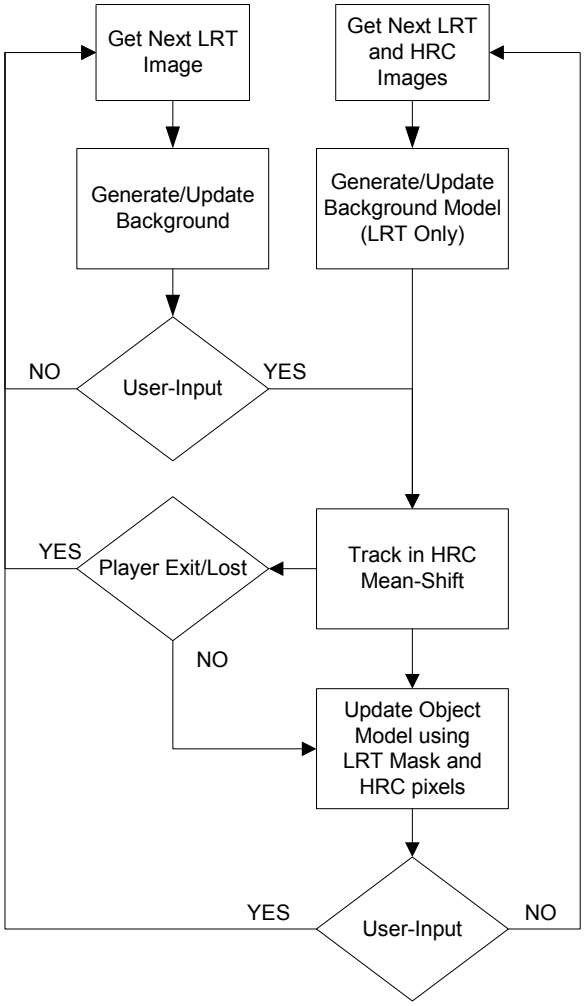
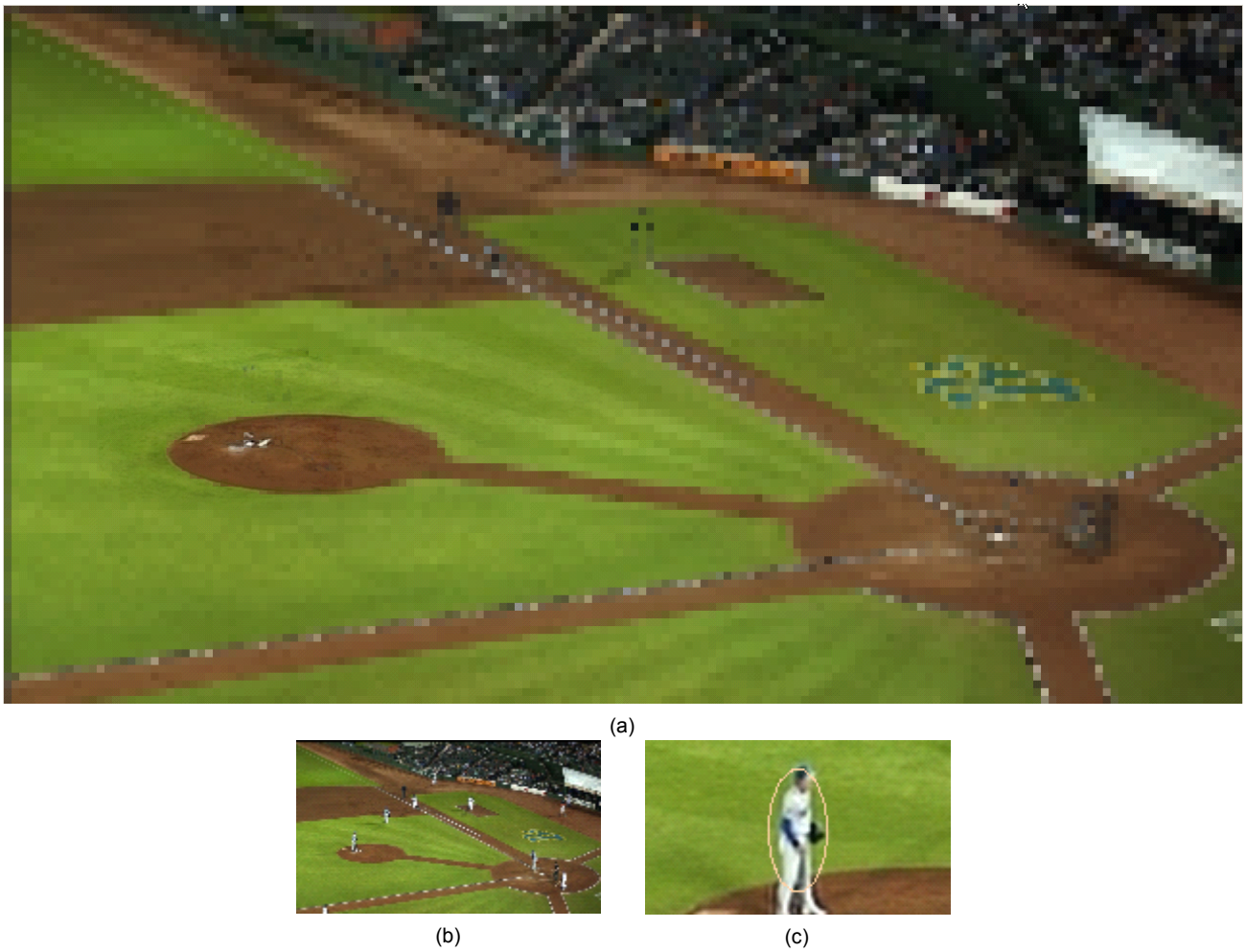


Figure 2. Flow chart of the algorithm to process LRT and HRC images from camera. It also highlights the role of semi-automatic player initialization and collaborative tracking.

3.1. System-Camera Interaction

The interaction between HD camera and our collaborative tracking system is shown in Fig. 1. The camera internal buffer stores HD frames at each time instant. The dotted horizontal line in the center of the figure shows the boundary between the camera internal buffer and what it shares with the system outside. The dotted rectangle on HD image shows hypothetical region requested by our system and to be delivered by the camera (after an expected delay of a few frames). The LRT image delivered by the camera reaches our system as next frame, but the HRC image requested by the system might arrive after a finite amount of delay. Please note that background update is performed in LRT to enable semi-automatic player initialization and to assist Mean-shift tracking in HRC image. The object model, however is updated in the HRC view only, as more pixels are available in this view and thus the object model can be updated more confidently with the help of background-foreground mask. An example of various image components of the system is shown in Fig.

3. The high-resolution background image at the same resolution as original HD image is shown in (a). An LRT image is shown in (b), while an HRC image is shown in (c) after detection and tracking.



(a)

(b) (c)

Figure 3. Collaborative HD player detection and tracking on base-ball sequence (1280x720). (a) HD background maintained from the low resolution background and individual high resolution images. (b) Low resolution thumbnail (LRT) image of the whole FOV. Please note the very small object sizes. (c) EPTZ high resolution cropped (HRC) image of a detected player (pitcher) being tracked.

3.2. System Architecture

A software operational scenario of our system is shown in Fig. 2 in terms of algorithm flow-chart. Our system starts by grabbing the LRT image of the scene. At this time, background generation is performed, which spans a few frames accumulating the scene background statistics. An automatic player initialization is possible with this approach, but the semi-automatic approach relying on end-user input is preferred. The main reason for this approach is that the end-user gets to select the player they want to see more closely in a field with tens of potential players to be tracked.

For tracking in the high-resolution view, we use mean-shift algorithm because of its real-time performance. One problem with the mean-shift algorithm for tracking is that the mean-shift kernel requires sufficient object overlap between successive frames to be tracked. There are two issues in our application domain that exacerbate this problem. Firstly, since the HRC image comes from a narrow FOV, the high-speed players being tracked spend very short amount

of time in this view. Secondly, there is a finite but non-negligible time lag Δt between the time that exact HRC bounding box is requested from the camera and the time when it is delivered to the outside system from camera. This time lag could be as high as around 10 frames. These issues call for a tracking approach that is more tolerant to object motion on the scale that object could leave the width of the spatial kernel entirely resulting in tracking error. Under these constraints, the original mean-shift algorithm will not provide good tracking in this view. This problem is overcome by our multi-kernel mean-shift algorithm with background information. Since multiple search kernels are initialized for object localization within a radius, the multi-kernel approach is found to be a lot more tolerant to frame delays and high-speed motions in narrow FOV images.

Once tracking has been performed, bject model update is handled in the HRC tracker since we have a lot more pixels to update the object model at high resolution. At the time of each update, the oldest samples of each pixel of the template (at D^{th} slice) are replaced with new ones. Based on foreground segmentation, template pixels corresponding to background pixels in current frame are not updated. Finally, the scene background image generated through LRT image is updated based on new HRC information and object location.

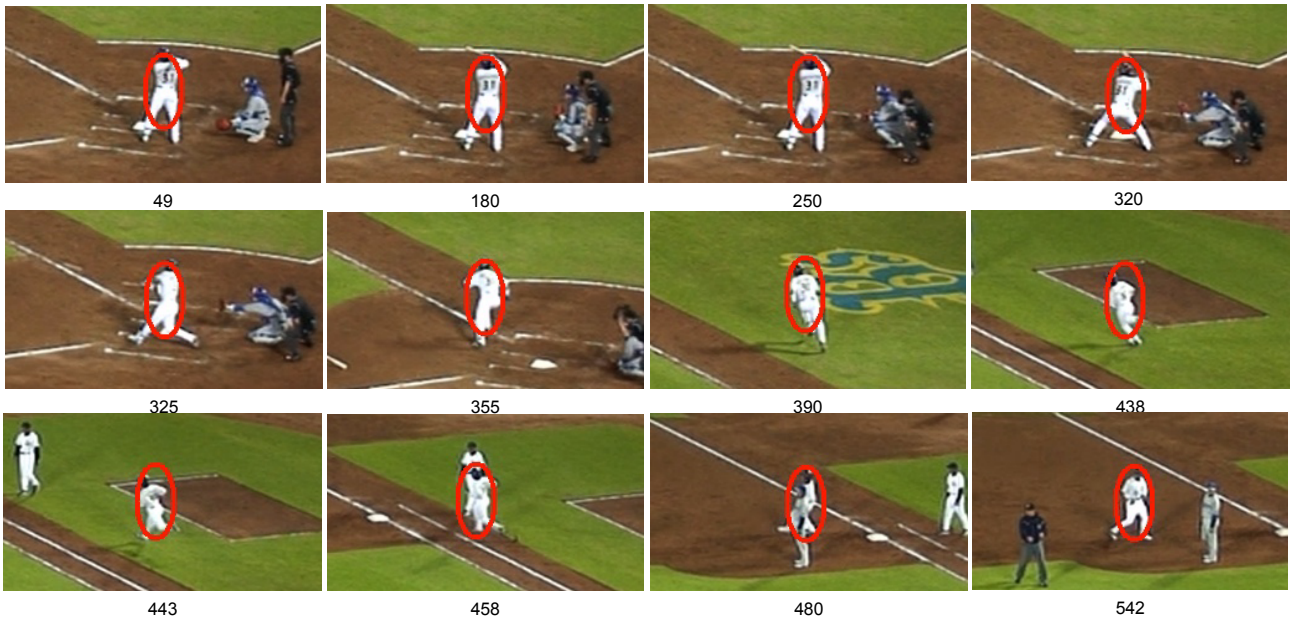


Figure 4. Collaborative HD player tracking in HRC view on base-ball sequence (1280x720). EPTZ tracking result images for high resolution display at end-user side with corresponding frame numbers.

4. RESULTS

We have tested our collaborative object detection and tracking system on a few video sequences. Results are presented on an HD video sequence example from the EPTZ camera capturing baseball game. Experimental results from an outdoor video sequence are also presented. The baseball sequence shows a lot of background noise due to audiences constantly moving, cameras flashing and other time varying illumination changes. Also, several scenes of occlusion are present which coupled with the fact that uniforms of several players appear the same present major problem in robustly tracking the object over time. Fig. 4 shows HRC images from a player being tracked. Please compare the amount of high-resolution information in these images with that of Fig. 3(b). It is apparent that our collaborative tracking approach presents a lot sharper details on the tracked target albeit at the same computational cost of dealing with much less amount of data. Please note also the robustness of tracking system to object shape deformations (frame 325), occlusions (frames 458 and 480) and scale changes (between frames 49 and 443).

Results from outdoor video sequence are presented in Fig. 5. The high-resolution background image is displayed in (a) which is the same size as HD image in camera. A low resolution thumbnail image is shown in (b). Different images from a tracking situation are shown in (c). All images are shown to the scale.

Figure 5. Collaborative HD human tracking in outdoor environment. (a) HD background image maintained through low resolution background and individual high resolution cropped images. (b) Low resolution thumbnail (LRT) image of the whole FOV. (c) EPTZ tracking result images for high resolution display at end-user side with corresponding frame numbers.

This video sequence also underlines the robustness of collaborative tracking system to changes in object scale as object moves farther from the camera, as well as severe occlusion. The automatic detection system based on background generation is also robust to gradual and sudden illumination changes due to weather conditions to a certain extent. Finally, we present the results of system performance in terms of average processing times per frame. These results are reported in table 1 for our collaborative detection and tracking system. For comparison, we also report the results of processing the original frames in HD resolution. As can be seen from this table, the collaborative tracking framework in EPTZ scenario, results in performance improvement of more than an order of magnitude.

Table 1. Average per frame processing times for tracking in our collaborative solution as compared to HD only tracking.

	Collaborative LRT+HRC	**HD Only**
Background Update	50 mSec.	800 mSec.
Tracking	20 mSec.	25 mSec.
Miscellaneous	5 mSec.	10 mSec.
Total	**75 mSec.**	**835 mSec.**

5. SUMMARY AND CONCLUSIONS

In this paper, we have addressed the issue of detecting and tracking objects of interest from HD video sequences. The approach is motivated by modern HD cameras with special mode of operation to conserve transmission and processing bandwidth. In this mode of operation, the camera transmits a low resolution thumbnail image of the whole filed of view at significantly less resolution as compared to the HD image it captures. In conjunction with that, these cameras provide a high resolution cropped image from a significantly less FOV. This electronic pan-tilt-zoom (PTZ) setting effectively lets the camera perform as a combination of a wide FOV static camera and a narrow FOV active camera unit. We present a background generation and object tracking system based on this operational scenario for high frame-rate object detection and tracking. Experimental results are reported on an HD sequence from baseball video, and outdoor video sequence. Future work will focus on computationally efficient means for generating and updating high-resolution background. Also, the issue of tracking in low-resolution in conjunction with high-resolution tracking needs to be explored.

ACKNOWLEDGMENTS

The authors would like to thank Oncel Tuzel for helpful insight and implementation.

REFERENCES

1. C. J. Needham and R.D. Boyle, *Tracking Multiple Sports Players Through Occlusion, Congestion and Scale,* British Machine Vision Conference, Manchester, UK, 2001.
2. F. Rafi, S. M. Khan, K. Shafiq and M. Shah, *Autonomous Target Following by Unmanned Aerial Vehicles*, SPIE Defence and Security Symposium 2006, Orlando FL.
3. F. Bashir, W. Qu, A. Khokhar and D. Schonfeld, *HMM-based Motion Recognition System using Segmented PCA*, IEEE International Conference on Image Processing, Genoa, Italy, 2005.
4. W. Qu, D. Schonfeld and M. Mohamed, *Distributed Bayesian Multiple Target Tracking in Crowded Environments using Multiple Collaborative Cameras*, *EURASIP J. Applied Signal Processing.* 2007 (In print).
5. S. M. Khan and M. Shah, *A Multiview Approach to Tracking People in Crowded Scenes using a Planar Homography Constraint*, 9th European Conference on Computer Vision ECCV 2006, Graz, Austria, 2006.
6. X. Zhou, R. T. Collins, T. Kanade and P. Metes, *A Master-Slave System to Acquire Biometric Imagery of Humans at Distance*, ACM International Workshop on Video Surveillance, Nov. 2003.
7. J. Migdal, T. Izo and C. Stauffer, *Moving Object Segmentation using Super-Resolution Background Models*, Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras, Oct. 2005.
8. C. Stauffer and E. Grimson, *Adaptive Background Mixture Models for Real-Time Tracking,* in Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Fort Collins, CO, Vol. II, 1999, pp. 246-252.
9. A. Elgammal, D. Harwood and L. Davis, *Non-parametric Model for Background Subtraction*, in Proc. European Conference on Computer Vision, Dublin, Ireland, Vol. II, 2000, pp. 751-767.
10. F. Porikli and O. Tuzel, *Bayesian Background Modeling for Foreground Detection*, ACM International Workshop on Video Surveillance and Sensor Networks (VSSN), Nov. 2005, pp. 55-58.
11. F. Porikli and O. Tuzel, *Multi-Kernel Object Tracking*, IEEE International Conference on Multimedia and Expo, July 2005, pp. 1234-1237.
12. D. Comaniciu, V. Ramesh and P. Meer, *Kernel-Based Object Tracking*, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol. 25, No. 5, pp. 564-575, 2003.