Open Journal of Applied Sciences, **2013, 3, 79-84**
Published Online March 2013 (http://www.scirp.org/journal/ojapps)

Large-scale Surveillance System based on Hybrid Cooperative Multi-Camera Tracking

Xiao Yan/s per[1], Dan Xu/s per[1], Bo Yao/s per[1,2]
[1]School of Information Science and Engineering, Yunnan University, Kunming, China
[2]School of Computer Science and Electronic Engineering, University of Essex, Colchester, United Kingdom
Email: xxiaoyen@gmail.com

Received 2012

ABSTRACT

In this paper, we proposed an optimized real-time hybrid cooperative multi-camera tracking system for large-scale automate surveillance based on embedded smart cameras including stationary cameras and moving pan/tilt/zoom (PTZ) cameras embedded with TI DSP TMS320DM6446 for intelligent visual analysis. Firstly, the overlapping areas and projection relations between adjacent cameras' field of view (FOV) is calculated. Based on the relations of FOV obtained and tracking information of each single camera, a homography based target handover procedure is done for long-term multi-camera tracking. After that, we fully implemented the tracking system on the embedded platform developed by our group. Finally, to reduce the huge computational complexity, a novel hierarchical optimization method is proposed. Experimental results demonstrate the robustness and real-time efficiency in dynamic real-world environments and the computational burden is significantly reduced by 98.84%. Our results demonstrate that our proposed system is capable of tracking targets effectively and achieve large-scale surveillance with clear detailed close-up visual features capturing and recording in dynamic real-life environments.

Keywords: Embedded Systems; Distributed Smart Cameras; Cooperative Visual Tracking

1. Introduction

The last few years have witnessed a widespread of smart cameras [1] in public places for surveillance purposes. However, it remain huge challenge for traditional surveillance system based on the framework of single camera and stationary cameras since the task of automated surveillance for public locations, which are usually crowded and wide-area, such as public transport stations, by using independent smart cameras almost impossible due to the limitation of cameras' field of view [2] and the heavy target occlusion problems [3]. Hence, scene surveillance using a cooperative multi-camera network [4] is becoming the preferred solution for surveillance camera users, as will not require major hardware upgrade. Therefore, a hybrid multi-camera tracking system based on embedded smart cameras including stationary CCTV cameras and moving PTZ cameras is introduced in this paper, specifically in our proposed system, stationary cameras are used for continuous and wide-area monitoring to detect events in important spots or in high places, and once abnormal events are detected via the large-scale view from the fixed cameras, PTZ camera is used for long-term tracking to obtain a close-up capture of the target and record the detailed information and features. In this way, by using our proposed framework, a visual surveillance system for large-scale monitoring with detailed close-up visual information capturing is constructed. Nevertheless, due to the computing limitation of the processing unit in each smart camera, advanced video processing algorithms which are usually of high computational complexity, cannot be performed without optimization. Consequently, in this paper, a novel hierarchical optimization paradigm with several practical techniques for common optimization is presented. According to our results, the performance is remarkably boosted using our proposed optimization methods.

The rest of this paper is organized as follows. In section 2, we provide the proposed multi-camera tracking system. Section 3 presents the proposed hierarchical optimization methodology. Section 4 presents the experiments and results and finally the conclusions are presented in section 5.

2. The Proposed Multi-Camera Tracking System

In order to cooperatively track and monitor moving targets in large-scale view together with detailed and close-up view, hybrid multiple cameras including stationary CCTV cameras and moving PTZ cameras with overlapping FOV are utilized to observer wide-area surveillance scenes from different views.

The block diagram of the proposed system is shown in **Figure 1**. Firstly, in the initial stage of our system, multi-camera calibration based on the ASIFT control points [5] for image (background) mosaic, as shown in **Figure 2**, is done to gain the image plane correspondence relationship between the adjacent cameras by homography transformation [6] for target hand-off. After that, multi-objects tracking is performed continuously in each single smart camera to obtain the trajectories of the moving targets. Based on trajectories from single camera tracking and the homographies obtained, trajectory transformation is carried out to hand-off moving targets between two adjacent views by computing projection error for multi-camera tracking. PTZ cameras without tracking object are calibrated using the uniform homography based calibration paradigm as the stationary cameras. Once a target is detected by a fixed CCTV camcamera, PTZ camera tracking is performed by tracking a fixed-size template from the target. The location of the template is initially obtained by transforming the id of the moving object tracked in the stationary camera using homography.

The entire embedded system is consisted of two components, a CCD color sensor providing NTSC or PAL video is used for capturing raw video data, and an embedded video analysis agent which is designed by employing a DaVinciTMS320DM6446 [8] dual-core device with an ARM9 and C64+ DSP

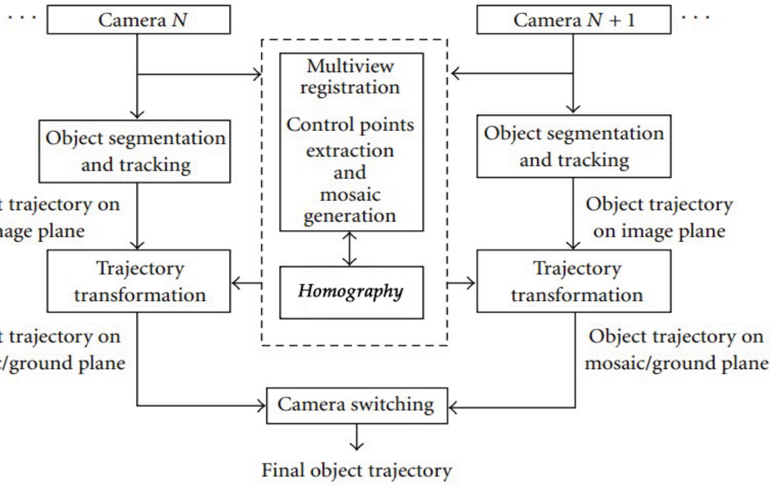

Camera N
Camera N + 1
Multiview registration
Control points extraction and mosaic generation
Homography
Object segmentation and tracking
Object segmentation and tracking
Object trajectory on image plane
Object trajectory on image plane
Trajectory transformation
Trajectory transformation
Object trajectory on mosaic/ground plane
Object trajectory on mosaic/ground plane
Camera switching
Final object trajectory

Figure 1. Block diagram of the proposed hybrid multi-camera tracking system.

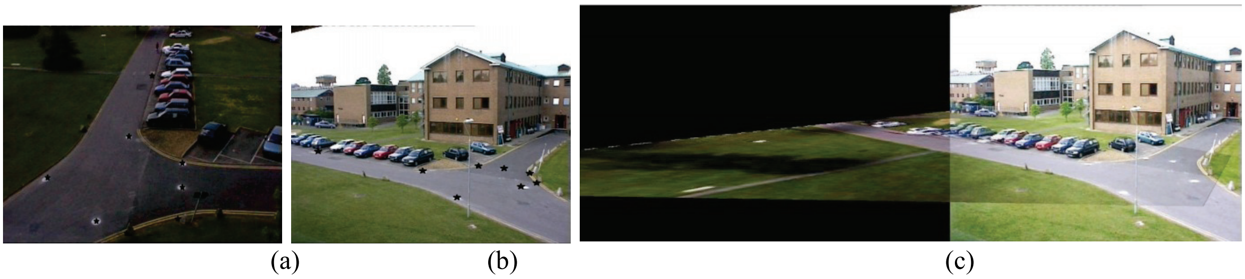


Figure 2. Multi-camera calibration. (a)Two camera views with corresponding control points (highlight with dark landmarks) on the overlapping FOV area; (c) image mosaic generated after homography transformation.

2.1. Single Stationary CCTV Camera Tracking

In multi-camera network surveillance, single stationary camera tracking is the fundamental module to obtain the information of the moving targets such as position, motion trajectory, shape, etc. Therefore, in our system, we utilized the tracking paradigm in [6]. Specifically, *Gaussian Mixture Models* (*GMM*) is employed to compute the background images of the surveillance scenes. Then, foreground objects (*Blob*) extraction is done to gain the bounding boxes and centroids of the moving targets. Finally, object tracking is performed based on *Mean-shift* and *Kalman* filter to analyze the motion history and trajectories.

2.2. Single Moving PTZ Camera Tracking

Once a moving target enters the overlapping FOV between the PTZ camera and the fixed camera, a robust target tracking algorithm [7] is carried out to continually track a fixed-size (48×48) template. The initial location of the template is obtained by transforming the centroid coordinate of the moving target tracked in the stationary camera based on homography. Experiment result is shown in **Figure 3** below, as can be seen, the target is successfully tracked in the moving PTZ camera for close-up capture of the clear and detailed features.

2.3. Multi-Camera Tracking

When moving objects enter the overlapping area between adjacent cameras including the stationary camera and static PTZ camera without target tracking, ground plane homography mapping is employed to create the viewpoint correspondence by mapping and matching target centroid positions between neighboring cameras, which is defined as follows:

$$\begin{bmatrix} x_i' \\ y_i' \\ 1 \end{bmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \times \begin{bmatrix} x_i' \\ y_i' \\ 1 \end{bmatrix} \qquad (1)$$

where *H(h_{11}~h_{33})* denotes *3×3* homography matrix describing the projection relationship of the two cameras while $\left(x_i\ , y_i\right)$ and $\left(x_i', y_i'\right)$ represent the corresponding centroids of the moving targets in each camera.

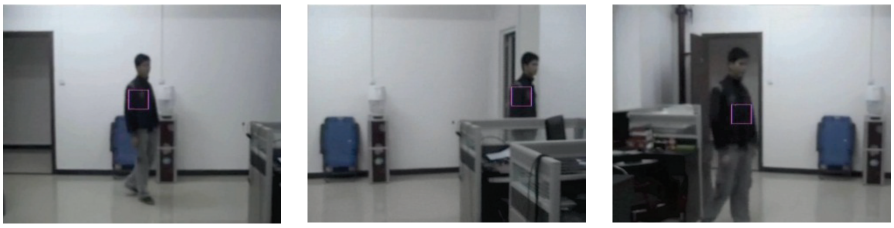
To calculate the homography matrix, in the initial stage of our system, we extract four best pairs of feature points from background images of the two adjacent surveillance scenes by using *ASIFT* [5] which is robust in dynamic real-world environments. Finally, based on the feature points extracted, *Levenberg-Marquardt* (*L-M*) [8] is performed to compute the homography with *Projection Error* (*PE*) minimization equation defined as follows:

$$PE = \sum_i \sqrt{\left(\frac{h_{11}x_i + h_{12}y_i + h_{13}}{h_{31}x_i + h_{32}y_i + 1} - x_i'\right)^2 + \left(\frac{h_{21}x_i + h_{22}y_i + h_{23}}{h_{31}x_i + h_{32}y_i + 1} - y_i'\right)^2} \qquad (2)$$

Then, based on the homography, target hand-off is done by examining *PE* between the centroid of two target candidates. If $PE = D^T$, where D^T is a parameter which can be dynamically set, then the two candidates are corresponding targets and marked the bounding box and trajectory in a unique color, shown in **Figure 4**.

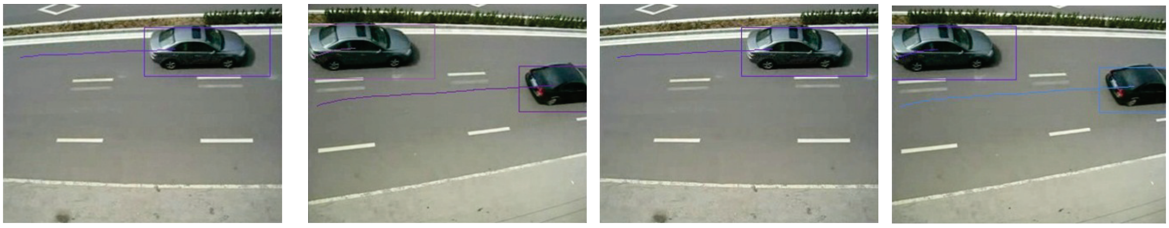
3. DSP performance optimization

To achieve the real time performance of the embedded DSP system, in this paper, a hierarchical optimization method is proposed based on DM6446 are used. According to the performance evaluation done by Code Composer Studio (CCS) profiling module, main per-



(a) **(b)** **(c)**

Figure 3. Experiment result of moving PTZ tracking.



(a) (b) (c) (d)

Figure 4.Target hand-off for multiple camera tracking, (a)(b) before corresponding; (c)(d) after corresponding.

formance bottlenecks are found, which are GMM for background reconstruction and moving object (*Blob*) extraction, respectively. Therefore, the proposed hierarchical optimization method is primarily concentrated in these two modules using project-level optimization, algorithm-level optimization and code-level optimization.

3.1. Project-level optimization

To maximize C/C++ compiler performance, the DSP code can be optimized comprehensively by using proper compiler setting [9]. Firstly, software pipelining is used to schedule instructions in a loop so that multiple instructions of the loop are executed in parallel. In C6000 compiler, we use “*-o2*” and “*-o3*” compiler options to arrange software pipelines for the codes automatically. Then, “*-pm*”, “*-mt*” and “*-op3*”compiler settings are employed in our compiler to reduce the performance cost in loop iterations. Additionally, to boost the efficiency, in the proposed system, the data which is frequently visited and processed are stored in internal DSP memory and important functions and procedures are executed in CACHE which is supreme fast memory.

3.2. Algorithm-level optimization

Since the detailed information of moving target such as texture, shape, etc. is not significantly essential for GMM and blob extraction, therefore before these two procedures, input video signal can be down-sampled to a smaller resolution for computational complexity reduction. So in our system, we resized the input video from D1 (*720×576*) to CIF (*360×288*) using resizer module in Video Processing Subsystem (VPSS) which is a standalone peripheral device on DM6446 for resizing video without any computational cost in DSP. Then, the resized video is analyzed by GMM and blob extraction to obtain the positions and bounding boxes of the moving objects. After that, the gained positions are re-mapped to D1 coordinate for the on-going procedures such as tracking, classification and recognition. Furthermore, for the purpose of better pipelining the algorithm to achieve higher performance, we divide the GMM function into three stages include background model initialization, updating and comparison, and performed separately. And after our algorithm-level optimization, according to the profiling result, the performance is greatly enhanced since the software pipeline is generated successfully.

3.3. Code-level optimization

Generally, the generation of software pipeline is a key step for code-level optimization. However, there are several common situations hinder producing software pipelines, i.e., loops nesting, in-loop function calling, jump instructions etc. Therefore, we examined and divided large loops into small loops to increases instruction-level parallelism guaranteeing the effective creation of software pipeline by using the instruction “*MUST_ITERATE*” in our system. After that, as the pixel value is *8-bit* length, to further improve the quality of the software pipeline, we utilized data packing techniques to pack and parallel process multiple pixels in one *32-bit* pack by executing the packing and unpacking instructions such as *“_memd8_const”, “_packl4”, “_hi”, “_lo”, “_subabs4”, “_cmpgtu4”, “_itoll”*, etc.

By utilizing our proposed hierarchical optimization method, the system performance is increased significantly as described in the **Table 1**.Since the performance of DSP core is at a clock rate of *810* MHz (*810* million clock cycles per second), the overall computational budget is reduced by *98.84%* and the system performance is boosted from *2.03* frames per second to *30* frames per second reaching the maximum frame rate.

Table 1.System performance comparison by using our proposed hierarchical optimization method

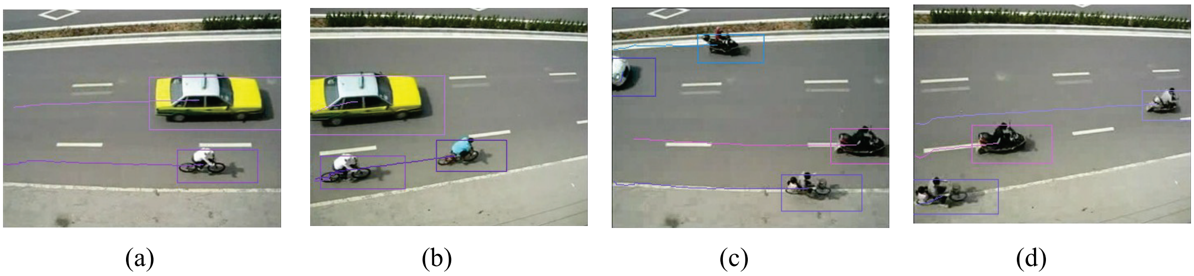
Module Name	**Before Optimization**		**After Optimization**		**Optimization Ratio**
	clock cycles	frames/s	clock cycles	frames/s	clock cycles
GMM Background Reconstruction	379672557	2.13 f/s	3387548	30 f/s	99.11%
Moving Object (Blob) Extraction	19218288	30 f/s	1249312	30 f/s	93.49%
Overall Performance	398890845	2.03 f/s	4636860	30 f/s	98.84%

4. Experiments and results

To demonstrate the robustness and effectiveness of our proposed system, we have built a test-bed environment around our campus by deploying multiple distributed cameras and performed several real-world experiments in various environments. Experiment results for cooperative stationary cameras tracking are shown in **Figure 5** (a) (b) (c) (d), the images demonstrate two result set of our multi-camera tracking system in an outdoor scene. **Figure 5** (a) (c) are the snapshots from the left camera view, while **Figure 5** (b) (d) are from the right camera view. Specifically, real-time multiple objects tracking is perform continuously on each individual camera to analyze the trajectories and bounding boxes of the moving targets, as displayed in **Figure 5**, each surveillance target is tracked successfully and marked with bounding box and trajectory in unique color to distinguish from others. After that, once moving targets enter the overlapping area between two adjacent cameras, object hand-off procedure is carried out to compute the accordance relationships of the targets in the overlapping area for ti-camera long-term tracking. As depicted in **Figure 5**, targets in the overlapping area are successfully tracked and hand-off and marked in their unique tracking color.

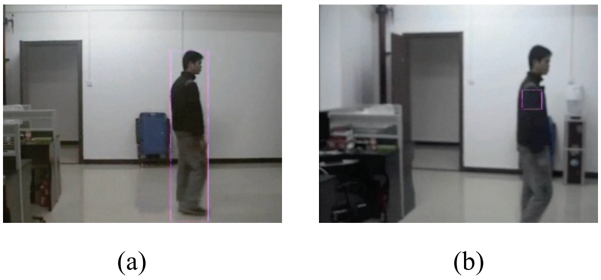
In **Figure 6**, experiments results for hybrid multi-camera tracking including stationary CCTV camera and moving PTZ camera is displayed. **Figure 6** (a) shows the tracking result from a stationary CCTV camera and accordingly **Figure 6** (b) demonstrates the result for moving PTZ camera tracking. As can be seen, the target is tracked successfully in bounding boxes with a same color.

Full demo video for our proposed embedded hybrid multi-camera tracking system is provided [10].



(a) (b) (c) (d)

Figure 5. Experiment results for dual camera-tracking in an outdoor environment around our campus



(a) (b)

Figure 6. Snapshot of our experiment devices and results in general view

5. Conclusions

In this paper, we represented the hybrid multi-camera tracking system using stationary CCTV cameras and moving PTZ cameras to address the problem of wide-area surveillance with close-up capture for detailed and clear visual cues. Our system is implemented on the embedded platform TI dual-core TMS320DM6446 (ARM+DSP). In our proposed system, a traditional tracking paradigm for single stationary camera tracking based on GMM, Mean-shift and Kalman filter is utilized to detect and track multiple targets. After that, when the target enters the overlapping area between adjacent cameras including stationary camera and PTZ camera, a homography based target hand-off procedure is performed for multi-camera tracking. Then, to obtain the close-up capture of the clear and detailed feature information from the target, large-scale long-term automate surveillance is achieved by utilizing a template-matching based PTZ camera tracking algorithm. However, the computation complexity for multi-camera system is huge especially for embedded processor based system. Therefore, to conquer the challenging problems in low-cost, reliable and efficient way, we proposed a novel hierarchical optimization method. The overall experimental results demonstrate the robustness and real-time efficiency and the stability of the embedded platform in dynamic real-world environments and the computational burden is significantly reduced by 98.84%.

REFERENCES

[1] R. Kleihorst, B. Schueler, A. Danilin, Architecture and applications of wireless smart cameras(networks), in: Proceedings of the IEEE International Conference on Acoustics, Speech, andSignal Processing, 2007.

[2] B. Rinner, W. Wolf, Introduction to distributed smart cameras, in: Proceedings of the IEEE96 (10).

[3] W. Hu, T. Tan, L. Wang, S. Maybank, A Survey on Visual Surveillance of Object Motion andBehaviors, in: IEEE Transactions Systems, Man and Cybernetics 34 (2004) 334–352.

[4] R.T. Collins, A.J. Lipton, H. Fujiyoshi, T. Kanade, Algorithms for cooperative multi-sensorsurveillance, in: Proceedings of the IEEE, 89 (2001) 1456–1477.

[5] Morel, J., Yu, G.: Is SIFT scale invariant? Inverse Problems and Imaging 5(1), 115–136 (2011).

[6] H. Aghajan and A. Cavallaro, Multi-camera Networks: principles andapplications. USA: Elsevier, 2009.

[7] R. Chaudhry, G. Hager, and R. Vidal, “Dynamic template tracking and recognition,” Arxiv preprint arXiv:1204.4476, 2012.

[8] M.I.A. Lourakis, A brief description of the Levenberg-Marquardt algorithm implemented bylevmar.

[9] Texas Instruments, Davinci-DM644x Evaluation Module technical reference”, March, 2006.

[10] http://youtu.be/9Xnm01Dk1LY.