International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:1, No:5, 2007

Model-Based Person Tracking Through Networked Cameras

Kyoung-Mi Lee and Youn-Mi Lee

***Abstract*—** This paper proposes a way to track persons by making use of multiple non-overlapping cameras. Tracking persons on multiple non-overlapping cameras enables data communication among cameras through the network connection between a camera and a computer, while at the same time transferring human feature data captured by a camera to another camera that is connected via the network. To track persons with a camera and send the tracking data to another camera, the proposed system uses a hierarchical human model that comprises a head, a torso, and legs. The feature data of the person being modeled are transferred to the server, after which the server sends the feature data of the human model to the cameras connected over the network. This enables a camera that captures a person's movement entering its vision to keep tracking the recognized person with the use of the feature data transferred from the server.

***Keywords*—** person tracking, human model, networked cameras, vision-based surveillance

I. Introduction

THE preceding studies on vision-based human tracking and monitoring are limited in that they use a single camera or, even in the case of multiple cameras, they apply three-dimensional human modeling to mainly recognize human body posture. This is a case where many individual eyes exist and each of these eyes can be seen as a collection of eyes that monitor people using a camera. Due to the limited view of a single camera, the tracking on an individual camera is possible but the image tracked by a camera cannot be transferred to another camera to be used in continued tracking for wide-area monitoring. To make up for this shortcoming, therefore, we need to develop a monitoring system that allows several cameras networked to monitor different areas while at the same time automating the tracking of a person spotted on one camera across a series of cameras.

Networked monitoring systems that enable cameras to communicate with each other, however, have not been studied sufficiently so far despite their benefits that they allow monitoring without the constraints of time and space that can be found in existing site monitoring systems. Javed *et. al* established correspondence across multi camera that determine the FOV line of each camera as viewed in other camera with real-time [2]. Porikli and Divakaran proposed the object-wise semantics from non-overlapping cameras and solved an inter-camera color calibration problem by using color histogram to determine inter-camera radiometric mismatch and correlation matrix [3].

In this paper, we propose a way to track multiple persons with data transfer between cameras via the network. Section 2 explains the system configuration and the data transfer. Section 3 describes the process of tracking persons with networked cameras, using a human model. The results of the experiments and conclusions are explained in Section 4.

II. Networked Camera System

The purpose of the proposed human tracking system is to use networked cameras with non-overlapping view that facilitate the acquisition of broad view and connect them over a network with a computer, while letting these cameras transfer the data on a tracking target. The feature data of a person who is recognized by a camera are sent over to another camera that is connected via a server. For this purpose, the cameras communicate data with each other in real time through a server, while the server needs to allow controls over data transfer between cameras and the input control by users. The server forms a network so that the cameras connected to it can recognize and exchange data with each other in real time. Each camera recognizes the data on a tracking target and its movements, and sends its feature data over to the server. Modeling of data on the tracking target is of utmost importance in order to send and receive data on the tracking target. We design a human model to exchange data on a tracking target over a network connection between cameras and a server. The human model contains data on the corresponding including the person's position, color, direction of movement, central point, and boundaries that form a human body [5].

Human tracking starts by initializing a human model when a person being tracked first appears on a camera. The human model designed is sent to the server and cameras so that they can use the data to track the target and authenticate the identity of the tracking target. The human model information is registered in a personlist that stores information on a person recognized by the server, which can be used in the data communication between the cameras and the server. The information on human model stored in the personlist changes frame by frame as the person moves, so that the model is updated for each frame to minimize any loss from data changes caused by the tracking target and thereby to facilitate more

This work was supported by Korea Research Foundation Grant (KRF-2004-003-D00376).

K.-M. Lee is with the Department of Computer Science, Duksung Women's University, Seoul 132-714, Korea (phone: +82-2-901-8348; fax: +82-2-901-8341; email: kmlee@duksung.ac.kr).

Y.-M. Lee is with the Department of Computer Science, Duksung Women's University, Seoul 132-714, Korea (phone: +82-2-901-8155; fax: +82-2-901-8341; email: blanchia@duksung.ac.kr).

precise tracking. The server sends out the feature data regarding the person who needs to be tracked to the cameras over the network. With the data from the server, a camera checks if a person who enters its field of vision is the one to be tracked and if so, initiates the tracking.

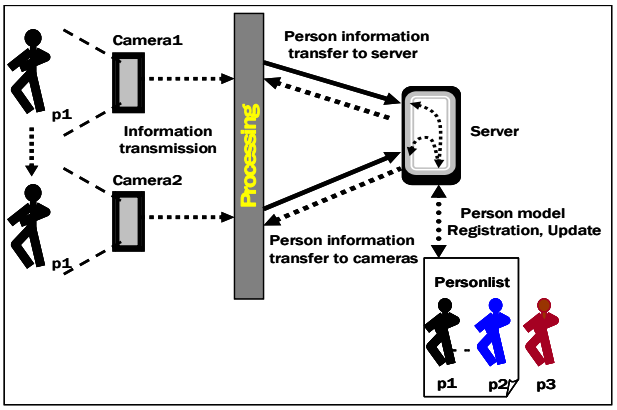


Fig. 1 The proposed networked camera system

Fig. 1 shows how multiple cameras transfer human-model data to one another by making use of the human model designed for this research. The person extracted by the first camera is stored in a personlist on the server by taking account of personal data (position, color, and relations to preceding frames, directions of movement). When the person enters the view of the second camera that is connected to the server, the camera checks if the person is registered in the personlist. If she is the one on the list, the second camera keeps on tracking the person. If not, it determines that she is a new person entering the network of cameras and registers a new human model on the personlist. This way each camera can check if a person entering its field of vision, or video frame, is one registered on the list or if she is a new person, or if the one who has been here and away is coming back. With these continued checks and tracking, the cameras that are distributed over a wide area can be used to make an end-to-end tracking of a person.

Fig. 2 The input images are in indoor environments (upper) and in outdoor environments (lower). The left images are acquired by camera 1 and the right images by camera 2.

III. Human Tracking through Networked Cameras

In this section, we introduce an approach that tracks persons through networked cameras. During tracking, the proposed approach compensates illumination noises, separates persons, models the separated persons, checks whether the persons were previously viewed, and sends the tracked information. Fig. 2 shows images that are acquired by each camera.

A. Adaptive background subtraction

Video image frames taken by a camera have variation in illumination conditions caused by lighting, time of day, and so on. Since noises by such conditions make tracking difficult, the noise should be removed from the image frame. To separate noises from images, an intrinsic image can be used to get a noise image by subtracting from the image frame. While adding the noise image to the image frame corrects background effectively, it is not sufficient to correct non-background objects. Recently, Matsushita *et. al* proposed a method for time-dependent intrinsic image estimation [4]. In this paper, we update a noise image frame-by-frame to estimate a time-varying intrinsic image [6]. We first initialize a noise image by subtracting the first image frame from the intrinsic image and then update the noise image frame-by-frame. If a pixel is similar with a noise pixel, the pixel is updated.

To detect moving persons after illumination correction, background subtraction provides the most complete feature data. In this paper, we build the adaptive background model, using the mean and standard deviation of the background [6]. Whenever a new frame arrives, a change in pixel intensity is computed using Mahalanobis distance to classify background or foreground (moving persons). The evaluated distance is compared to a difference threshold previously observed from the sequence of images. If a pixel is classified to background, the adaptive background model is updated with the pixel. Fig. 3 shows result images after illumination correction and background subtraction of Fig. 2.

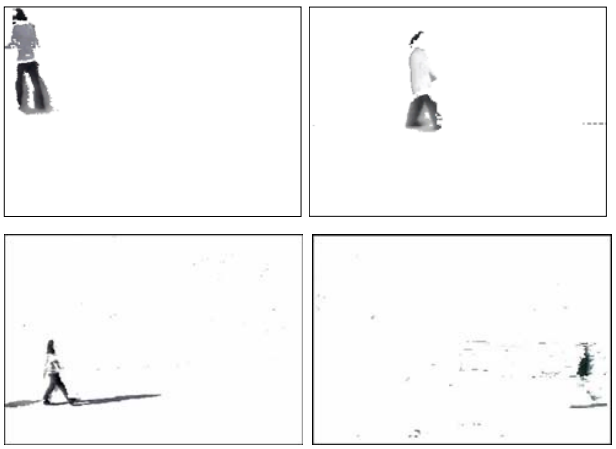


Fig. 3 Background subtraction with illumination correction of Fig.2

B. Person model initialization

After background subtraction (Fig. 4(b)), tracking persons should be initialized when they start to appear in the video. To

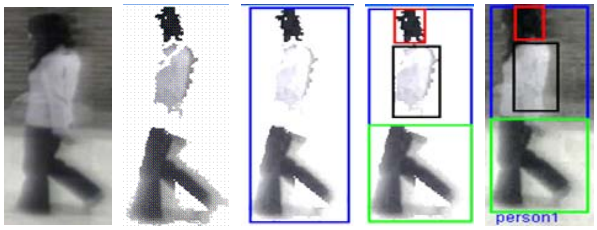
group segmented foreground pixels into a blob and to locate the blob on a body part, we use a connected-component algorithm which calculates differences between intensities between a pixel and its neighborhoods and then merge small blobs into large blobs and neighboring blobs that share similar colors are further merged together to overcome over-segmentation generated by initial grouping. Each blob contains information such as an area, a central position, color, position, a bounding box, and a boundary to form a human body. Then, created blobs are removed according to criteria, such as, too small, too long, or too heterogeneous incompact blobs.

As a person can be defined as a subset of blobs, which correspond to human body parts, blobs in a frame should be assigned to corresponding individuals to facilitate multiple individual tracking. Let P_0 be a subset of blobs B_i. The set of potential person areas is built iteratively, starting from the P_0 set and its adjacent graph. The set P_1 is obtained by merging compatible adjacent blobs of P_0. Then, each new set of merged blob P_k is obtained by merging the set of merged blob P_{k-1} with the original set P_0. Finally, a candidate person CP_n contains the built sets of merged blob P_k, i.e., $CP_n = \bigcup_{k=0}^{K} P_k$, n=1…N where N is the number of persons.

To match blobs to body parts, we use a hierarchical person model. The high level of the model contains a whole person model and its information (Fig. 4(c)). The person model is defined by three body parts and their geometrical relations in the middle level (Fig. 4(d)), and as blobs and their geometrical and color relations in the low level as follows:

$$CP_n = \left(R_n^0, \{C_n^1, R_n^1\}, \{C_n^2, R_n^2\}, \{C_n^3, R_n^3\}\right) \quad (1)$$

where R^0 means a relation among three parts. C_n^j and R_n^j mean a set of blobs and their relationships of the j-th body part of CP_n, respectively.



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

Fig. 4 A hierarchic human model: (a) original image, (b) background subtraction image, (c) person area, (d) 3 body parts(head, torso, legs), and (e) result

C. Adaptive Person Modeling

Since a person is tracked using a person model (Fig. 4(e)), person model information is stored to track multiple-persons. Even though the total motion of a person is relatively small between frames, large changes in the color-based model can cause simple tracking to fail. To resolve this sensitivity of color-model tracking, we compare the current blobs to a reference person model. The reference person model should be compensated according to occlusion as well illumination changes.

Let a person CP_n^{t-1} represented by an average (μ_n^{t-1}) and a deviation (σ_n^{t-1}), which are computed up to time t-1 and new blobs B_i^t and their relations Br_i^t are formed in frame t. The minimum difference between the person model CP_n^{t-1} (Eq. (1)) and the new blobs B_i^t is computed as follows:

$$d_n^t = \min_{j=1\cdots3}\left(\frac{\left\|B_i^t - \mu_n^{t-1,C^j}\right\|_p}{\sigma_n^{C^j}}\right) + \min_{j=0\cdots4}\left(\frac{\left\|Br_i^t - \mu_n^{t-1,R^j}\right\|_p}{\sigma_n^{R^j}}\right) \quad (2)$$

where μ_n^{t-1,C^j} and μ_n^{t-1,R^j} mean a set of averages of blobs and relations in the j-th body part at time t-1, respectively. σ_n^{t-1,C^j} and σ_n^{t-1,R^j} a set of deviations of blobs and relations, respectively. If the minimum distance is less than a predefined threshold, the proposed modeling algorithm adds blobs B_i^t and relations Br_i^t to corresponding adaptive person model (μ_n^{t-1,C^j} and μ_n^{t-1,R^j}) and updates the adaptive model by recalculating their center and uncertainties. Here, similarity thresholds are set empirically and can be adjusted by a user.

D. Model-based person tracking

Tracking people poses several difficulties, since the human body is a non-rigid form. After forming blobs, a blob-based person tracking maps blobs from the previous frame to the current frame, by computing the distance between blobs in consecutive frames. However, such a blob-based approach for tracking multiple persons may cause problems due to the different number of blobs in each frame: blobs can be split, merged, or even disappear or be newly created.

To overcome this situation, many-to-many blob mapping can be applied [1]. In this paper, we assume that persons CP_n^{t-1} have already been tracked up to frame t-1 and new blobs B_i^t are formed in frame t. Multi-persons are then tracked as follows:

Case 1: If B_i^t is included in CP_n^{t-1}, the corresponding blob in CP_n^{t-1} is tracked to B_i^t.

Case 2: If a blob in CP_n^{t-1} is separated into several blobs in frame t, the blob in CP_n^{t-1} is tracked to one blob in frame t and other blobs at time t are appended into CP_n^{t-1}.

Case 3: If several blobs in CP_n^{t-1} are merged into B_i^t, one blob in CP_n^{t-1} is tracked to B_i^t and other blobs are removed from CP_n^{t-1}.

Case 4: If B_i^t is included in CP_n^{t-1} but the corresponding blob does not exist, B_i^t is added to CP_n^{t-1}.

Case 5: If B_i^t is not included in CP_n^{t-1}, the blob is considered as a newly appearing blob and thus a new person is added to the person list.

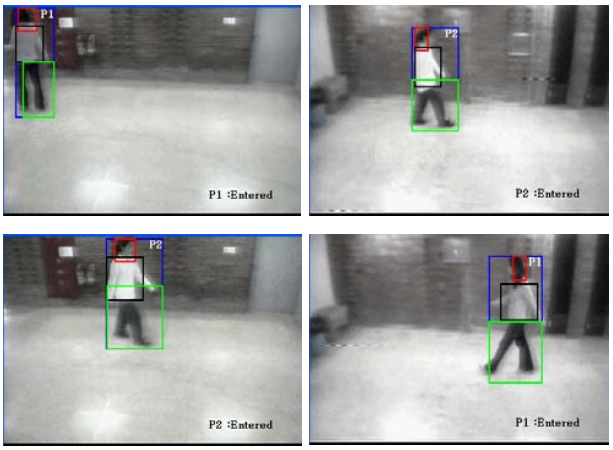
where including a region into a person with a bounding box means the region overlaps over 90% to the person. Corresponding a blob to the adaptive person model is computed using Eq. (2). In addition to simplify the handling of lists of persons and blobs, the proposed approach can keep observe

existing persons exiting, new persons entering, and previously monitored persons re-entering the scene. One advantage of such a model-based tracking is to relieve the burden of correctly blobbing. Even though a blob can be missed by an illumination change, model-based tracking can retain individual identity using other existing blobs.

IV. Results and Conclusion

The proposed method is implemented with JAVA (JMF) on the Microsoft Windows 2000 XP platform. The experiment was carried out on a computer using the images (420×316) acquired from two UNIMO CCN-541 security cameras.

Fig. 5 and 6 show the results of the human tracking with the cameras 1 and 2 in Fig. 2. The persons initially tracked by camera 1 (person 1) and camera 2 (person 2) have moved over time to the field of vision of camera 2 (person 1) and camera 1 (person 2). Each camera sends over to the server the modeling data it created for the features of the person it acquired. Upon receiving the data, the server delivers the feature data regarding the human model to all the cameras connected to the network, so that each camera keeps on tracking the recognized person whose movement enters its FOV.



Camera 1 Camera 2

Fig. 5 Result images in indoor environments

Camera 1 Camera 2

Fig. 6 Result images in outdoor environments

To evaluate the tracking performance of the proposed algorithm, we used tracking accuracy which divides the number of correctly tracked persons by the number of tracked persons (Fig.7). The proposed tracking algorithm is achieved 91% of person 1 and 97% of person 2 in the indoor environments, and 82% of person 1 and 80% of person 2 in outdoor environments.

The proposed human tracking method with networked cameras has been experimented in an indoor setting, like a long hallway, and an outdoor setting, like a long path. The future works of this research are to implement a virtual space that will provide an in-depth look into the trails of the tracked persons' movements and to develop an intelligent interface.

References

[1] S. Park and J. K. Aggarwa, "Segmentation and tracking of interacting human body parts under occlusion and shadowing," in *Proc. of International workshop on motion and video computing*, pp.105-111, 2002.

[2] Javed, Z. Rasheed, O. Alatas and M. Shah, "KNIGHT^{M}: A real-time surveillance system for multiple overlapping and non-overlapping cameras," in *Proc. of the International conference on multimedia and expo*, 2003.

[3] F.M.Porikli and A. Divakaran, "Multi-camera calibration, object tracking and query generation", in *Proc. of the International conference on multimedia and expo*, pp. 653-656, 2003.

[4] Y. Matsushita, K. Nishino, K. Ikeuchi and M. Sakauchi, "Illumination normalization with time-dependent intrinsic images for video surveillance," *IEEE trans. on PAMI*, 26(10):1336-1347, 2004.

[5] S. Khan and M. Shah, "Consistent labeling of tracked objects in multiple cameras with overlapping Fields of View," *IEEE trans. on PAMI*, 25(10):1355-1360, 2003.

[6] K. M. Lee and Y. M. Lee, "Tracking multi -person robust to illumination changes and occlusions," in *Proc. of International of conference on artificial reality and telexistence*, pp. 429-432, 2004.

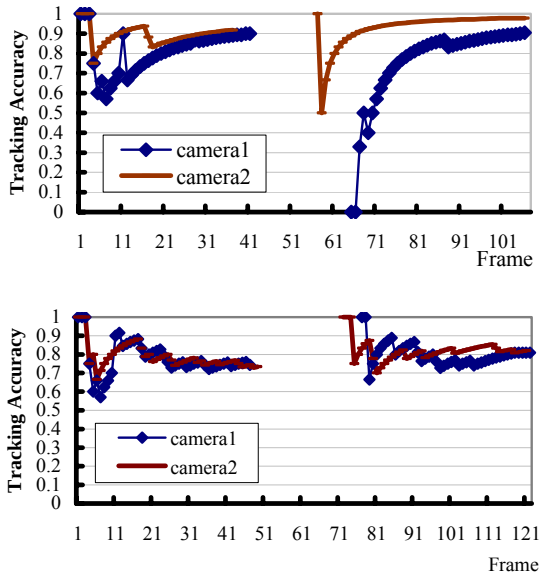


Fig. 7 Tracking accuracy on camera1 and camera2. The upper one is an indoor result and the lower one an outdoor result.