Tracking of Multiple Objects Across Multiple Cameras with Overlapping and Non-Overlapping Views

LiangJia Zhu, Jenq-Neng Hwang
Department of Electrical Engineering, Box 352500
University of Washington, Seattle, WA, USA
{zhulj, hwang}@u.washington.edu

Hsu-Yung Cheng
Dept. of Computer Science and Information Engineering
National Central University, Jhongli, Taiwan
chengsy@csie.ncu.edu.tw

***Abstract*—In this paper, we propose a fully automated approach for tracking of multiple objects across multiple cameras with overlapping and non-overlapping views in a unified framework without initial training. For single camera cases, Kalman filter and adaptive particle sampling are integrated for multiple objects tracking. When extended to multiple cameras cases, the relations between adjacent cameras are learned systematically by using image registration techniques for consistent handoff of tracking-object labels across cameras. In addition, object appearance measurement is employed to validate the labeling results. Experimental results demonstrate the performance of our approach on real video sequences for cameras with overlapping and non-overlapping views.**

I. INTRODUCTION

Surveillance tracking of multiple video objects across large area requires camera networks work cooperatively for reliable handoff of tracked objects from one camera to another. If single camera tracking is effective, the main task for tracking across cameras is to establish the correspondence between the tracks of the same object when it is seen in a new view [1].

In earlier work [2], we have presented a single camera tracking approach by integrating Kalman filter and adaptive particle sampling for multiple video object tracking. This approach is very robust to occlusion and segmentation errors. In this paper, we take one step further to track objects across multiple cameras. Two typical scenarios for multiple cameras tracking are considered, one is for cameras with overlapping field of views (FOVs), and the other is with non-overlapping (disjoint) FOVs, i.e. "blind" area, as shown in Figure 1. We assume that prior knowledge of camera network topology is given, since it can be obtained easily and brings several advantages, such as decreasing the computation complexity [3]. For two adjacent cameras, the FOV lines for both overlapping and non-overlapping cases, which will be briefly described in Section IV-A, can be automatically established by using effective image registration techniques without using training video sequences as in [1]. Our approach also uses the appearance measurement in single camera tracking to validate the labeling result, instead of using FOV lines as the only feature for consistent labeling [1]. Moreover, the labeling result is further used to continuously update the pre-computed FOV lines.

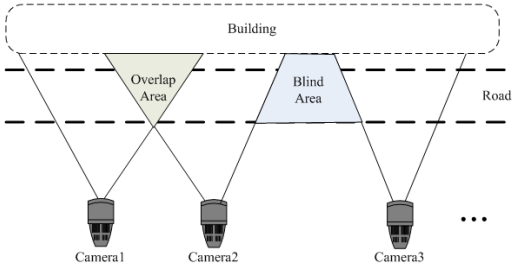

Figure 1. The configuration of camera network system.

The rest of this paper is organized as follows: In Section II, we review some of the related work in tracking across multiple cameras. Next, in Section III, we briefly describe the single camera tracking method. Then, Section IV-A gives the details about how to establish FOV lines for both overlapping and non-overlapping cases. Section IV-B shows the consistent labeling process across cameras. The experimental results are presented in Section V, followed by conclusion in Section VI.

II. RELATED WORK

The approaches for solving the consistent labeling problem for tracking across camera networks with overlapping FOVs can be roughly categorized into the following three schemes: (1) *The Feature Matching Scheme* is a common strategy for consistent labeling which matches geometrical or other features of tracked objects directly. The FOV lines, which delineate the region of one camera that is visible in another camera, have been used for consistently labeling [1]. Instead of using training videos, a projective invariants based method was proposed in [4], which manually selects corresponding feature points from two camera images to establish FOV lines. Multiple features, such as epipolar geometry, homography, landmark, apparent height and color, can also be combined using a Bayesian Network for better similarity measure [5]. (2) *The 3D Information Based Scheme* achieves the consistent labeling by projecting the location of each object in the world coordinate system, and establishing equivalence between objects from various camera views, if camera calibration and 3D site model are created [6]. (3) *The Alignment Scheme* achieves consistent labeling by aligning different camera views through geometrical transformation, e.g., the transformed motion trajectories of objects as observed in

978-1-4244-3828-0/09/$25.00 ©2009 IEEE

different cameras and computing plane homographies for each match [7].

On the other hand, tracking targets across multiple cameras with non-overlapping FOVs requires the solving of a correspondence problem by learning the relations between non-overlapping cameras. In [8], the concept of FOV lines is extended to non-overlap views. Different camera views can be transformed to a common ground plane, and tracking in the "blind" region is achieved by motion prediction [9]. However, the feature correspondence in transforming camera views is manually selected. In [3], spatio-temporal and appearance cues are integrated to constrain correspondences. In [10], the topology of camera network is learned and the gap between non-overlapping cameras is bridged by providing the probabilistic estimation of the location and time with which a target may re-appear.

III. Single Camera Tracking

We have proposed a robust multi-target tracking scheme for single camera scenarios [2]. Before applying the tracking algorithm, a background model is estimated and updated from video to segment video objects (VOs) from background. The segmented object is represented by an ellipse. Kalman filter is applied to track the object. The system state is $[x, y, \dot{x}, \dot{y}, a, b]^T$, where (x,y) is the object centroid position, $(\dot{x}, \dot{y})$ is its velocity, and (a,b) are the length of ellipse major axis and minor axis, respectively. The measurement state is defined as $[x, y, a, b]^T$. After Kalman filter prediction, proper measurement is selected and an enhanced version of probabilistic data association (EPDA) [11] is utilized to associate the measurement with each target object for filter update.

A. Adaptive Particle Sampling under Segmentation Error

In measurement selection, if there is no occlusion or segmentation error, the segmented VOs are reliable and the measurement for a target object can be obtained by referring to the segmented VOs. Otherwise, adaptive particle sampling is performed to find reasonable measurement candidates instead. In order to discover segmentation errors, a validation gate is built upon the predicted system state of each target object. If no measurement within the validation gate of a target object can be found, then we regard it as a segmentation error because either the positions of all the VOs are far away from the predicted position, or the major axes and minor axes of the ellipses fitted on the VOs are very different from the predicted ones for the target object. A segmentation error can also be discovered when a segmented VO is within the validation gate but there is a big difference between the size of the VO and the size of the tracked object.

B. Adaptive Appearance Update under Occlusion

Occlusion are discovered when two or more tracked objects start to merge with one another. Occlusion detection can be accomplished by checking the predicted state of every pair of target objects in the tracking list. When the degree of occlusion for an object exceeds a threshold, adaptive appearances should be applied to ensure the robustness against occlusion [2]. Also, in this case, the appearance update of the object should be stopped. It was shown in [2] that applying the proposed adaptive appearances based on color matrices and color histograms significantly improves the positioning and scaling accuracy compared to non-adaptive appearances under serious occlusion.

IV. Tracking Across Multiple Cameras

In order to consistently label new objects entering a camera, the relations between adjacent cameras are first estimated offline by applying image registration techniques. After that, the brightness difference is compensated by matching the histograms of overlapping area. The brightness difference will be also used to modify the histogram of an object when it passes across cameras. Both the distance to FOV lines and appearance are used to measure the degree of matching between each candidate object correspondence pairs to select the best global label.

A. Establish Field Of View (FOV) Lines

The FOV of a camera C_i is a rectangular pyramid in the space. The intersection of each planar side s of this pyramid with the ground plane defines 3D FOV lines [1], which mark the viewing limit of C_i from side s. If the viewing coverage of camera C_i is overlapping with that of C_j from side s, there would be a 2D line $L_j^{i,s}$ that marks the limit of what may be visible in the image plane of C_j, and this line is called an *FOV line*. At most, there are four FOV lines of one camera as visible in the other one, which correspond to four sides of image plane.

Therefore, if we stitch two images I_i and I_j from overlapping cameras C_i and C_j, the overlapping area will mark the common visible area from both cameras, and the intersection of the four sides of I_i with I_j are the approximation of FOV lines of C_i in C_j. Two images I_i and I_j can be aligned automatically through the following steps.

1) Extracting Landmark Points

The landmark points in each image are extracted by using scale invariant feature transformation (SIFT) [12]. First, candidate landmark points are extracted by searching for the scale-space extrema over all scales and locations using difference of Gaussian function. Once a candidate point is found, a 3D quadric function is fitted to refine the landmark locations. Then, one or more orientations are assigned to each landmark point based on local image gradient information. Finally, orientation histograms are computed to represent all those landmark points.

2) Finding Matching Landmark Points

After obtaining landmark points in each image, the corresponding pairs are indexed using an efficient nearest neighbor search method, called Best Bin First algorithm [13]. It is a *k-d* tree based algorithm with a modified search order by taking into account the position of the query location, i.e., the bins in feature space are searched in order of increasing

distance from the query location. A good approximation can still be found by cutting off further search after a certain number of the nearest bins have been explored.

3) Aligning Two Images

Given a set of corresponding landmark points $(x, y) \leftrightarrow (x', y')$ from two images, the homography H that describes the projective transformation between those points can be estimated by the Random Sample Consensus (RANSAC) algorithm [14] in an iterative way. In each iteration *k*, four pairs of landmark points are randomly selected to calculate a candidate homography H_k. Then all other landmark points in image I_i are transformed with H_k, and the number of inliers in this iteration is counted as those with distance $d_H(H_k(x, y), (x', y')) < t_h$, where d_H is the Euclidean distance between two points and t_h is the tolerance threshold. After the iterative process is finished, the largest set of inliers are used to estimate the best homography H^* in a least square sense. Finally, image I_i is registered to I_j by transforming it with H^*. The FOV lines of C_i into C_j are approximated by the intersection between the transformed four sides of image I_i and image I_j.

For cameras with non-overlapping views, we assume that there is an intermediate image I_{ij} which is overlapping with both images I_i and I_j. This is a reasonable assumption for most surveillance scenarios where cameras views may not be overlapping, while taking some bridging images can be easily feasible. In this way, images I_i and I_j can also be registered sequentially in a similar way as in reconstructing panoramas. The extended FOV lines are defined as the extended virtual boundaries of FOV of one camera. One such extended FOV line example, which is created by the use of one intermediate image, is illustrated in Figure 4.

After registration, a visibility map V_i is built for image I_i, where $V_i^j(x, y) = 1$ represents that position (*x*,*y*) is also visible in camera C_j. The subset of all cameras *C* in which the *n-th* object O_i^n in C_i can be seen in other cameras is given by

$$C^i(n) = \{j \mid V_i^j(x, y) = 1, \forall i \neq j \text{ and } j \in C\}, \quad (1)$$

where (*x*,*y*) is the current feet position that denotes the lower end of the ellipse major axis fitted on the object.

B. Brightness Calibration of Neighboring Cameras

After image registration, the grayscale cumulative histograms of overlapping areas are calculated for both images. Histogram matching method [15] is used to compensate for brightness changes.

C. Consistent Labeling Across Cameras.

When the *n-th* object O_i^n enters C_i from a specific side *s*, the visibility map is first checked for the visibility of O_i^n in other cameras. If O_i^n is only visible in C_i, then assign a new global label to O_i^n. Otherwise, O_i^n should be also visible in other cameras defined in $C^i(n)$. Suppose all camera videos have been synchronized, the corresponding label of O_i^n is searched in the following steps.

First, a list of corresponding candidates is generated for O_i^n. For each $C_j \in C^i(n)$, search for objects O_j^m with $d(O_j^m, L_j^{i,s}) < t_d$ that moves toward the visible region of C_i from C_j from side *s*, where $d(O_j^m, L_j^{i,s})$ is the minimum distance between the feet position of O_j^m and FOV lines $L_j^{i,s}$ associated with the specific side *s* that O_i^n enters C_i. An object is defined to be moving toward the visible area of C_i, if it enters C_j from invisible area of C_i and has been tracked in C_j for a certain time, e.g., one second. In particular, if two cameras C_i and C_j have non-overlapping views, then the feet position of an object in the "blind" region is predicted by a constant velocity model in which the constant velocity is approximated by using least-square fitting of its previous visible feet positions in C_j.

According to the FOV constraints in [1], the candidate object with the minimum distance to its corresponding FOV line is selected as the most possible match for O_i^n. We define the likelihood of matching two objects in terms of the distance to FOV line as

$$P_d(O_i^n, O_j^m) = \frac{1}{\sqrt{2\pi}\sigma_d} \exp(-d^2(O_j^m, L_j^{i,s}) / \sigma_d^2), \quad (2)$$

where σ_d controls the width of the region considered for selecting candidate matches.

To take into account of the errors in detecting feet position and estimating FOV lines, we also define the likelihood of matching two objects in terms of appearance using color dissimilarity as

$$P_c(O_i^n, O_j^m) = \frac{1}{\sqrt{2\pi}\sigma_c} \exp(-d^2(h_1, h_2) / 2\sigma_c^2), \quad (3)$$

where h_1, h_2 are the color histograms of the two objects O_j^m and O_i^n respectively. And $d(h_1, h_2)$ is the dissimilarity (in terms of Bhattacharyya distance) between the histograms.

Then, for each candidate, the likelihood of assigning the global label of O_j^m to O_i^n is defined as the product of P_d and P_c as

$$P(O_i^n, O_j^m) = P_d(O_i^n, O_j^m) P_c(O_i^n, O_j^m). \quad (4)$$

If the object with the highest likelihood as computed in (4) has been globally labeled, then we assign its global label to O_i^n. Otherwise, a new global label is assigned to both of them.

After handoff labels, the corresponding feet position in C_j is saved. With sufficient number of feet positions being collected, a least square line fitting method can be used to update FOV lines.

There is a special case about initial occlusion. If more than one candidates have very high and close likelihoods as computed in (2), it is possible that these candidates may be in occlusion initially when they enter C_i. In this case, a list of labels is created for O_i^n. If O_i^n splits later, then only the appearance likelihoods are computed between the objects splitted from O_i^n and that from the label list to select the most likely global labels.

V. Experimental Results

The proposed approach was tested for two cameras with overlapping and non-overlapping views separately. Figure 2 shows the results of two surveillance cameras monitoring of a parking lot with overlapping views. The left camera is camera 1, the right one is camera 2. Two tracked persons P1 and P2 in camera 2 were passing through the FOV lines almost at the same time (see Figure 2(c)). The approach can successfully label them after they entered camera 1. Also, it can be seen from Figures 2(c) and 2(d) that the FOV lines are established accurately since when P2 was passing through FOV line L_2^1, at the same time he appeared in camera 1. And as he was passing through L_1^2 in camera 1, he was leaving camera 2 in the meantime. The tracking results for non-overlapping case are shown in Figure 3, where two persons P1 and P2 were walking from right to left. P1 entered the blind region first but with a slower speed than that of P2. The established extended FOV line for non-overlapping case is given in Figure 4. Although the scene in the intermediate image had changed slightly since it was not taken at same the time as the testing sequence had been taken, the image registration and extended FOV line can still be effectively established. It can be seen from Figure 3(c), based on the predicted position approximated from the constant velocity, that P2 entered the left camera from the blind region earlier since it is closer to the extend FOV line,. The tracking results for both cases with occlusions are also given in Figure 2(b), 2(c) and Figure 3(d).

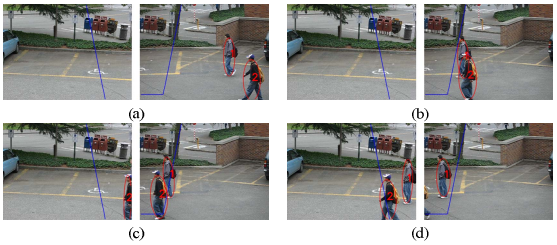

Figure 2. Tracking results for cameras with overlapping views. FOV lines are marked with blue color.

VI. Conclusion

We have presented an efficient way of tracking objects across multiple cameras for both overlapping and non-overlapping cases in a unified framework. The (extended) FOV lines can be established automatically without training. Combining the distance to FOV lines and color similarity to select correct matches makes our approach more applicable to initial occlusion scenarios and less sensitive to the errors in estimating FOV lines and feet position of objects being tracked. Updating FOV lines with tracking results further increases the adaptation capabilities of our approach.

Figure 3. Tracking results for cameras with non-overlapping views.

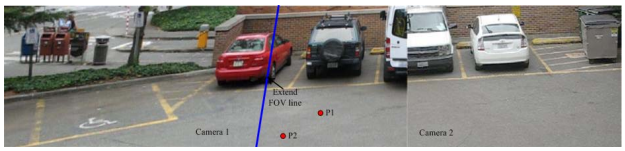

Figure 4. Motion prediction in "blind" region. The extended FOV line is marked with blue color. The predicted positions for P1 and P2 when P2 appears in camera 1 are marked with red color.

References

[1] S. Khan, M. Shah, "Consistent labeling of tracked objects in multiple cameras with overlapping fields of view", PAMI, vol. 25, no. 10, pp. 1355-1360, 2003.

[2] H.Y. Cheng, J.N. Hwang, "Resolving occlusion and segmentation errors in multiple video object tracking", in SPIE Conference on Computational Imaging, San Jose, 2009.

[3] K. Chen, C. Lai, Y. Hung, C. Chen, "An adaptive learning method for target tracking across multiple cameras", CVPR, pp. 1-8, 2008.

[4] S. Velipasalar, W. Wolf, "Recovering field of view lines by using projective invariants", ICIP, vol. 5, pp. 24-27, 2004.

[5] T. Chang, S. Gong, "Tracking multiple people with a multi-camera system", WOMOT with ICCV, 2001.

[6] Q. Cai and J.K. Aggarwal, "Tracking human motion in structured environments using a distributed camera system", PAMI, vol. 21, no. 11, pp. 1241-1247, 1999.

[7] L. Lee, R. Romano and G. Stein, "Monitoring activities from multiple video streams: establishing a common coordinate frame". PAMI, vol. 22, no. 8, pp. 758-767, 2000.

[8] O. Javed, Z. Rasheed, O. Alatas, and M. Shah, "Knight M: a real time surveillance system for multiple overlapping and non-overlapping cameras", ICME, vol. 1, pp. 649-652, 2003.

[9] P. Kumar, A. Chilgunde, S. Ranganath and W. Huang, "Multi-camera target tracking in blind regions of cameras with non-overlapping fields of view", BMVC, pp. 397-406, 2004.

[10] D. Makris, T. Ellis, and J. Black, "Bridging the gaps between cameras", CVPR, Vol. 2, pp. 205–210, 2004.

[11] H. Cheng, J. Hwang, "Multiple target tracking for crossroad tracking utilizing modified probabilistic data association", IEEE Int'l Conf. on ASSP, Honolulu, Hawaii, April, 2007.

[12] David G. Lowe, "Distinctive image features from scale-invariant keypoints", IJCV, vol. 60, no. 2, pp. 91-110, 2004.

[13] J Beis and D Lowe, "Shape indexing using approximate nearest-neighbour search. in high-dimensional spaces", CVPR, pp. 1000-1006, 1997.

[14] M. A. Fischler, R. C. Bolles. "Random Sample Consensus: a paradigm for model fitting with applications to image analysis and automated

cartography". Communications of the ACM, vol. 24, no. 6, pp. 381-395, 1981.

[15] U. Fecker, M. Barkowsky, and A. Kaup, "Improvingthe prediction efficiency for multi-view video coding usinghistogram matching", in Picture Coding Symposium (PCS 2006), Beijing, China, Apr., 2006.