COMBINED ESTIMATION OF CAMERA LINK MODELS FOR HUMAN TRACKING ACROSS NONOVERLAPPING CAMERAS

Young-Gun Lee, Jenq-Neng Hwang
Department of Electrical Engineering
University of Washington, Box 352500
Seattle, WA 98195, USA
{lygstj, hwang}@uw.edu

Zhijun Fang
School of Information Technology
Jiangxi University of Finance and Economics
Nanchang, Jiangxi 330032, P.R. China
zjfang@gmail.com

ABSTRACT

Human tracking across multiple cameras is highly demanded for large scale video surveillance. To successfully track human across multiple uncalibrated cameras that have no overlapping field of views, a system to train more reliable camera link models is proposed in this paper. We employ a novel approach of combining multiple camera links and building bidirectional transition time distribution in the process of estimation. Through the unsupervised scheme, the system builds several camera link models simultaneously for the camera network that has multi-path in presence of the outliers. Our proposed method decreases incorrect correspondences and results in more accurate camera link model for higher tracking accuracy. The proposed algorithm shows the effectiveness by evaluating in the real-world camera network scenarios.

Index Terms— human tracking, unsupervised learning, disjoint camera view, multiple cameras, camera link model

1. INTRODUCTION

Recently, video surveillance systems are getting deployed both in private and public areas for monitoring and security purposes. Due to the limited field of view (FOV) of a single camera, human tracking across multiple cameras is critically needed. In spite of plenty of research efforts being made on tracking multiple people across the uncalibrated cameras with disjoint views, this topic has remained quite an open and challenging issue, due to the potential changes of cameras' illumination, and variations of tracked person's perceived appearance between cameras from different perspective, lighting condition and viewpoint. To overcome these challenges, many researchers resort to solving object tracking across camera view problem based on the correspondences of tracks among multiple sets of candidates [1][2][3][4][5]. More specifically, in [3], they exploit an incremental scheme to model both the color variations and posterior probability distributions of spatio-temporal links between cameras for the tracking. This technique is completely unsupervised, however, they require a large amount of training data required in the incremental learning procedure. An adaptive method for learning spatio-temporal relationships is proposed in [4], which needs much less training data than that in [3]. However, the features for matching the correspondences are integrated either with uniform weights or empirically determined weights, resulting in worse performance. In [5], the camera link models are learned based on a fully unsupervised scheme in the presence of outliers. The weights for integrating multiple features are systematically determined during the training stage. All the above works individually estimate only one camera link model for a pair of two directly-connected cameras and apply it for both directions during the tracking. However, all the features used in the matching are not fully symmetrical, especially, transition time distribution from one camera to the other is usually not the same in the opposite direction.

In this paper, we present a novel estimation scheme to acquire more precise camera link models and show the efficiency of proposed algorithm. More specifically, we combine several cameras which are directly connected to jointly train the associated camera link models. It enables to decrease the ratio of outliers and wrong correspondences in the training stage. Further, we employ an asymmetric bidirectional transition time distribution, which is essential in improving the tracking performance of the camera networks.

The rest of this paper is organized as follows. In Section 2, we give an overview the overall tracking system. We provide the algorithmic details of the proposed scheme in Section 3. The experimental results are reported in Section 4, followed by the conclusion in Section 5.

2. SYSTEM OVERVIEW

The original unsupervised multi-camera tracking system proposed in [5] consists of two major stages, i.e., training and testing, as follows (see Fig. 1):

2.1. Training Stage

First, persons who leave or enter the FOV of each pair of

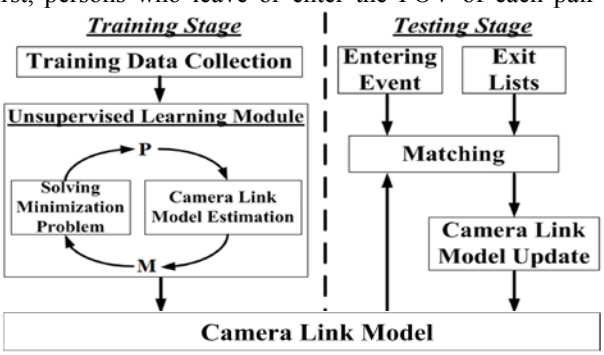

Figure 1. A multiple camera tracking system [5].

exit/entry zones in two directly-connected cameras are collected. The deterministic annealing and the barrier methods are applied to estimate the "symmetrical" camera link model between two directly-connected zones [5]. The camera link model contains several features: transition time distribution, holistic and regional brightness transfer functions (BTFs) [7], region mapping matrix, region matching weights and feature fusion weights [5]. These results are used on the following testing stage, assuming the same camera link model can be applied to either direction of human movement and re-identify the human who cross the cameras.

2.2. Testing Stage

All the multiple cameras perform the single camera human tracking by adaptive Kalman filtering and multiple kernels tracking with projected gradients [6]. Each camera C_i maintains an exit list $\mathbf{L}_{i,k}$ for each exit/entry zone k, which

$$\mathbf{L}_{i,k} = \left\{ O_{i,k}^1, O_{i,k}^2, ..., O_{i,k}^{N_{i,k}} \right\}, \tag{1}$$

contains all the observations $\{O_{i,k}\}$ of the persons who have left the FOV from zone k within training time T_{max} seconds from now, and $N_{i,k}$ denotes the number of observations on exit/entry zone k of camera i. When a person enters the FOV of the other connected camera, tracking algorithm finds the best correspondence among the exit lists by computing the matching scores. The matching distance between two observations, O^1, O^2, can be computed as the weighted sum of distances according to Eq. (2),

$$score = -\sum_{j=1}^{N_{feature}} \alpha_j \times feature\_dist_j(O^1, O^2), \tag{2}$$

where α_j is the weight derived from the training stage. When the lowest score on possible pairs is lower than a predefined threshold, we conclude that the pair is the same person. Otherwise, we will regard it as a new person. The re-identification results with higher confidence can also be further used to update the camera link models.

3. PROPOSED ALGORITHMS

We propose a new estimation method to produce more reliable camera link models for camera networks. Through combining several links in the training stage, each link can produce better camera link model between each directly connected camera pair. In this paper, we jointly train the pairwise camera link models based on a group of three zones, each of which has two different connected links for our proposed algorithms. The deterministic annealing is again employed [9][10] to build the optimum binary permutation matrix **P** and the corresponding camera link model between each pair of connected cameras.

3.1. Camera Link Model Estimation

In the training stage, we estimate the camera link model based on sets of observations acquired from directly-connected cameras. For example, camera C_1 is directly connected to camera C_2 in Fig. 2. An exit set **X** and an entry set **Y** are collected from C_1 and C_2, within training time T_{max}, respectively.

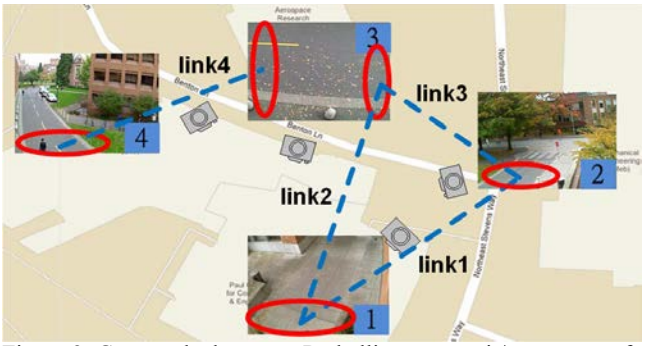

Figure 2. Camera deployment. Red ellipses are exit/entry zones for four links and the number in blue rectangle denotes the camera number.

$$\mathbf{X} = \left[x_1, x_2, ..., x_{N_1} \right], \quad \mathbf{Y} = \left[y_1, y_2, ..., y_{N_2} \right], \tag{3}$$

where x_i and y_j are exit and entry observations, and N_1 and N_2 are the number of the observations. Each element of **X** and **Y** maintains the exit or entry time stamp, holistic color, region color and texture features. We exploit these information to match the correspondences automatically between the exit/entry observation sets. Our goal is to determine the $(N_1+1) \times (N_2+1)$ permutation matrix **P**, whose element P_{ij} is set to 1 if x_i corresponds to y_j. Otherwise, it is set to 0. The extra column and row represent the outlier entries. Finally the camera link model can be derived from the estimated correspondence matrix **P**. The problem can be written as a constrained minimization integer programming problem:

$$\hat{\mathbf{P}} = \arg\min_{\mathbf{P}} J(\mathbf{P}), \tag{4}$$

$$s.t.\ P_{ij} \in \{0,1\} \quad \forall i \le N_1+1,\ j \le N_2+1, \tag{5}$$

$$\sum_{i=1}^{N_1+1} P_{ij} = 1 \quad \forall j \le N_2, \quad \sum_{j=1}^{N_2+1} P_{ij} = 1 \quad \forall i \le N_1, \tag{6}$$

$$\begin{aligned} J(\mathbf{P}) = {} & cost_{time} + cost_{holistic\ color} + cost_{region\ color} \\ & + cost_{region\ texture} + cost_{entropy} + cost_{outlier}, \end{aligned} \tag{7}$$

where J is the objective function to be minimized and comprises several cost terms, $\{cost_{feature}\}$, each of which stands for the distance of the associated feature between exit and entry observations [5]. The objective function J is iteratively minimized with **P** approaching to binary matrix. The converged matrix **P** is then used to derive the corresponding camera link model. The region color and region texture features are necessarily incorporated to overcome the issues introduced by the different viewing perspectives. As shown in Fig. 3(a), a target is divided into 6 regions based on the shown height ratios. Note that the exit observation in C_3 as shown in Fig. 3(c) can be re-identified to be the entry observation in C_2 as shown in Fig. 3(b), via the (holistic) color appearance of the whole body, as well the region color and texture appearance through appropriate region mapping and weightings.

3.2. Combined Camera Links Training

Nowadays more and more surveillance cameras are getting deployed on the streets, resulting in a spider web like connected camera links, i.e., one camera is directly connected with multiple other cameras. Most methods that use appearance transfer functions [4][5][7][10] build link

model by using 1-to-1 pair of connected cameras to represent the space-time relationship and color/texture transfer models. However, if there are several forked roads, many persons who leave one exit will not necessarily show up in one specific entry and they are going to be regarded as outliers in matrix **P**. The error increases as the number of outliers increases [8]. In other words, the estimation of **P** may make more incorrect correspondences, resulting in less accurate camera link models. Thus, we are motivated to combine several pairs of connected cameras during the estimation of the **P**'s to jointly create more reliable camera link models. Fig. 2 shows an actually deployed 4-camera network $\{C_1, C_2, C_3, C_4\}$ surrounding the Electrical Engineering building of the University of Washington, there are in total four camera links as denoted by blue dash lines. So the person who exits from C_1 can enter C_2 or C_3 by crossing the link1 or link2, vice versa for C_2 and C_3. The previous method in [5] treats a person who enters C_3 as an outlier during building camera link model for link1. On the other hand, this person becomes an inlier by combining link1 and link2 together.

Suppose we have an additional set, **Z**, from entry, C_3.

$$\mathbf{Z} = \left[z_1, z_2, ..., z_{N_3} \right], \tag{8}$$

where z_k is an entry observation and N_3 are the total number of observations within the training time T_{max}. By considering **Z** in the previous $\mathbf{P}_{\text{link1}}$ that denotes the correspondence matrix of link1, i.e., the correspondence is now extended from every exit observation of C_1 to every entry observation of both C_2 and C_3. The permutation optimization problem can now be rewritten as follows:

$$\hat{\mathbf{P}}^r = \arg\min_{\mathbf{P}^r} J(\mathbf{P}^r), \text{ where } \mathbf{P}^r = \left[\mathbf{P}_{\text{link1}} \quad \mathbf{P}_{\text{link2}} \right] \tag{9}$$

$$s.t.\ P_{ij}^r \in \{0,1\}, \quad \forall i \le N_1 + 1,\ j \le N_2 + N_3 + 1, \tag{10}$$

$$\sum_{i=1}^{N_1+1} P_{ij}^r = 1 \quad \forall j \le N_2 + N_3, \quad \sum_{j=1}^{N_2+N_3+1} P_{ij}^r = 1 \quad \forall i \le N_1, \tag{11}$$

where $\mathbf{P}^r$ is a concatenated matrix of $\mathbf{P}_{\text{link1}}$ and $\mathbf{P}_{\text{link2}}$, except the outlier row is merged to one row. The wrong matches occur when the solver is trapped in the local minimum during solving the minimization problem with deterministic annealing [9]. However, **Z** contains several inliers from exit zone of C_1, so incorporating elements of **Z** is equivalent to enhancing the magnitude of the global minimum in the optimization. Thus, incorrect exit/entry correspondences decrease as the percentage of the outliers in the training data is reduced. As a result, the resulting camera link models are more accurate due to more true positive pairs and less false positive pairs being utilized in the training.

3.3. Bidirectional Link Model

In [5], the camera link model is constructed based only on one exit/entry direction (e.g., exit from C_1 and entry on C_2) and that is applied to both directions of link1 (i.e., exit from C_1 and entry on C_2 as well as exit from C_2 and entry on C_1) to find correspondences during the testing. Here camera link model includes transition time distribution, brightness transfer function, region mapping matrix, region matching weights, and feature fusion weights. All of these components can be reversed except the transition time distribution, which is represented as a probability density function. So when we exploit one camera link model for tracking of both directions of the same link, it inevitably degrades the tracking accuracy. Transition time distribution can be similar only when the road's conditions, i.e., width, slope, surface, are similar on both directions, this is not the case most of time. Thus, bidirectional link models, i.e., asymmetric link models, have to be estimated in the training stage and employed in the testing stage.

4. EXPERIMENTAL RESULTS

We have experimented with several videos recorded on four cameras shown in Fig. 2. The FOVs of the cameras are disjointed and the cameras are not calibrated. We assume that we know the camera topology and the exit/entry regions in the FOVs of the cameras.

4.1. Camera Link Model Estimation

The deployed camera network has four links between five exit/entry zones as shown in Fig. 2. To build camera link models, we collect videos with persons' passing by any exit/entry zone for 15 minutes in the training stage. They have 245 distinct persons and 148 outliers. We compare the results of 1-to-1 pair-wise link modeling method with the result of 1-to-*N* combined link modeling method by using these datasets. In the 1-to-1 method, which is based on the pair of one exit corresponding to one entry zone, data from two directly connected cameras are utilized for training. On the other hand, 1-to-*N* method, which has one exit corresponding to several entry zones, utilizes data from combined links for training. Table I shows the number of outliers in each link and the comparison results in the training stage where OL denotes outlier and C_i–C_j denotes a link from exit C_i to entry C_j. By combining two links, outliers decrease in all the links, i.e., in C_3–C_2, outliers decrease as much as 45% (1-to-1 method has 67 outliers and 1-to-N method has just 37 outliers) . We evaluate the results by using the following three metrics: precision, recall and F_1 score as defined in Eq. (12). It can be seen in Table I that in

$$precision = \frac{TP}{TP + FP}, \quad recall = \frac{TP}{TP + FN}, \quad F_1 = \frac{2 \times precision \times recall}{precision + recall}. \tag{12}$$

almost all links, 1-to-*N* method obtains equal or more true positive (*TP*) and less false negative/positive (*FN*/*FP*) values than those of 1-to-1 method. In C_2–C_3, 1-to-1 method achieves more *TP*, however, also has more *FP*. In case of 1-to-1 method, the average of precision is 0.53, recall is 0.83 and F_1 score is 0.61. On the other hand, 1-to-*N* method achieves 0.98 in precision, 0.89 in recall and 0.93 in F_1 score. Thus, the proposed 1-to-*N* algorithm significantly outperforms 1-to-1 in every evaluation measure. In Fig. 4, we compare some of estimated camera link models, in terms of BTF and region mapping weights, by applying to Fig. 3(b) and 3(c). Euclidean distance of BTF transferred histograms is 0.0954 between blue and green, 0.0706 between blue and black, and 0.0659 between blue and red. Thus, the resulting camera link models estimated from 1-to-*N* method are more accurate. Fig. 5 shows the asymmetric

Table I. Experimental results in the training stage

Link	Method	OL	TP	FN	FP	precision	recall	F_1
C_1–C_2	1-to-1	41	18	6	11	0.62	0.75	0.68
	1-to-*N*	35	20	4	0	1	0.83	0.91
C_1–C_3	1-to-1	45	3	3	10	0.23	0.5	0.32
	1-to-*N*	21	4	2	0	1	0.67	0.80
C_2–C_1	1-to-1	19	6	1	5	0.55	0.86	0.67
	1-to-*N*	7	7	0	0	1	1	1
C_2–C_3	1-to-1	14	12	0	3	0.8	1	0.89
	1-to-*N*	7	10	2	1	0.91	0.83	0.87
C_3–C_1	1-to-1	49	2	0	13	0.13	1	0.23
	1-to-*N*	47	2	0	0	1	1	1
C_3–C_2	1-to-1	67	27	4	5	0.84	0.87	0.85
	1-to-*N*	37	31	0	1	0.97	1	0.98

(a) (b) (c)

Figure 3. (a) A target is divided into 6 regions based on the shown height ratios. (b) An exit observation in C_3. (c) An entry observation of the same person in C_2.

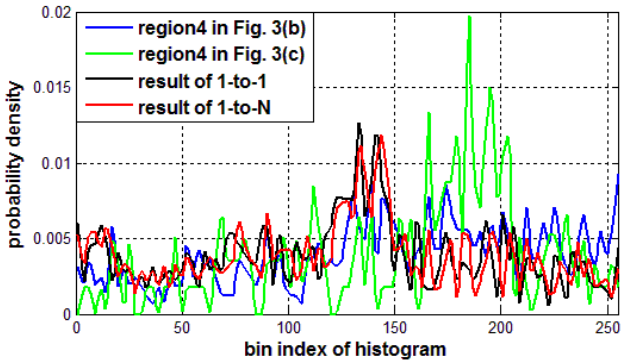

Figure 4. Histogram comparison. Blue: the histogram from region 4 in Fig. 3(b). Green: the histogram from region 4 in Fig. 3(c). Black: the histogram from Fig. 3(c) after applying BTF and region mapping derived from 1-to-1 method. Red: the histogram from Fig. 3(c) after applying BTF and region mapping derived from 1-to-N method. Note only one channel is shown here for demonstration purpose.

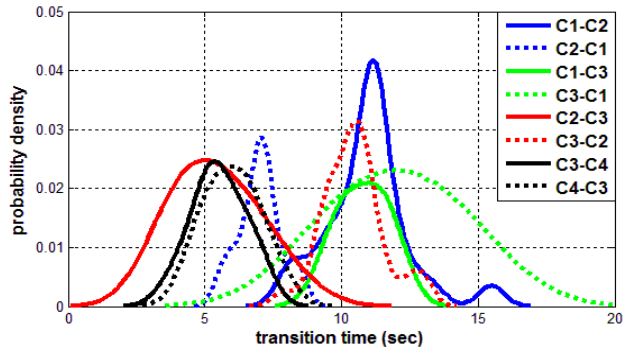

Figure 5. Asymmetric transition time distributions of all 8 links.

Table II. Tracking accuracy

Method	Unidirectional 1-to-1	Bidirectional 1-to-1	Unidirectional 1-to-*N*	Bidirectional 1-to-*N*
Accuracy	72.2%	74.7%	86.1%	87.3%

transition time distributions of eight links. The same color lines are used to represent the estimated results of same links of opposite directions, i.e., blue solid and dash lines are for link1. Even though black and green solid lines are similarly distributed to dash lines of the same color, blue and red solid lines are quite different from the same color dash lines, reflecting the transition time differences between bidirectional link1 and link3, i.e., gentle slopes occur in these two links. With these more precise transition time distributions, we are able to enhance the accuracy of time spent on both directions. It is also worthwhile to mention to the comparative computational cost involved in these two methods. The average total CPU times of the 1-to-1 and 1-to-*N* training algorithms are 220.718 sec and 578.897 sec over the 15-minute training videos of 4 cameras on Intel core i5 3.0 GHz with 6 GB of RAM. The testing time of both methods are almost the same.

4.2. Tracking Accuracy

To test the tracking accuracy after the training, we further use about 15 minutes of 4-camera videos, different from the training videos, which include 276 persons, 79 pairs with an outlier rate about 50.4%. Table II shows the results of tracking accuracy. We compare the performance based on four experiments. The unidirectional 1-to-1 method is the same as [5] and the bidirectional 1-to-*N* method is our proposed algorithm in this paper. Another two additional experiments are also performed to justify the effect of the proposed bidirectional scheme. As shown in Table II, the proposed method achieves the best accuracy of 87.3%. By combining multiple links in the camera link model estimation, the accuracy is improved about 13.9% and 12.6% on unidirectional and bidirectional method, respectively. As we expect, camera link models become more accurate, so we can correctly re-identify more pairs. Taking into account the bidirectional scheme contributes to enhance the accuracy about 2.5% and 1.2% in 1-to-1 and 1-to-*N* methods separately. The effect is less than the combining multiple links in the training, however, it can give larger advantages when the load's conditions are highly asymmetric.

5. CONCLUSION

In this paper, we presented a new estimation method to build camera link models for tracking human across the forked camera networks. Based on the proposed unsupervised scheme, we provide a solution that estimates accurate camera link models by matching the correct correspondences between the observations from one exit and several entries in the presence of outliers. To show the efficiency of our proposed method, we conducted experiments on real scenario videos. We compared our method with established models and found that ours performed favorably on both camera link model estimation and human tracking across the cameras.

6. REFERENCES

[1] O. Javed, Z. Rasheed, K. Shafique, M. Shah, "Tracking across multiple cameras with disjoint views," *Proc. IEEE 9th Intl. Conf. on Computer Vision*, vol. 2, pp. 952-957, Oct. 2003.

[2] Z. Wu, N. I. Hristov, T. L. Hedrick, T. H. Kunz, M. Betke, "Tracking a large number of objects from multiple views," *IEEE 12th Intl. Conf. on Computer Vision*, pp. 1546-1553, Sept.-Oct. 2009.

[3] A. Gilbert and R. Bowden, "Tracking Objects Across Cameras by Incrementally Learning Inter-camera Colour Calibration and Patterns of Activity," *Proc. European Conf. on Computer Vision*, pp. 125-136, May 2006.

[4] K. Chen, C. Lai, Y. Hung, and C. Chen, "An Adaptive Learning Method for Target Tracking Across Multiple Cameras," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition,* pp. 1-8, June 2008.

[5] C.-T. Chu, J.-N. Hwang, J.-Y. Yu, and K.-Z. Lee, "Tracking Across Nonoverlapping Cameras Based on The Unsupervised Learning of Camera Link Models," *ACM/IEEE Intl. Conf. on Distributed Smart Cameras*, pp. 1-6, Oct.-Nov. 2012 (also appeared in *IEEE Trans. on CSVT*, vol. 24, no. 6, pp. 979-994, June 2014).

[6] C.-T. Chu, J.-N. Hwang, S.-Z Wang, and Y.-Y. Chen, "Human Tracking by Adaptive Kalman Filtering and Multiple Kernels Tracking with Projected Gradients," *ACM/IEEE Intl. Conf. on Distributed Smart Cameras*, pp. 1-6, Aug. 2011.

[7] T. D'Orazio, P. Mazzeo, and P. Spagnolo, "Color brightness transfer function evaluation for non overlapping multi camera tracking," *ACM/IEEE Intl. Conf. on Distributed Smart Cameras*, pp. 1-6, Aug.-Sept. 2009.

[8] C.-T. Chu, J.-N. Hwang, Y.-Y. Chen, and S.-Z. Wang, "Camera Link Model Estimation in a Distributed Camera Network Based on The Deterministic Annealing and the Barrier Method," *Proc. IEEE Conf. on Acoustics, Speech and Signal Processing*, pp. 997-1000, Mar. 2012.

[9] H. Chiu and A. Rangarajan, "A new point matching algorithm for non-rigid registration," *Computer Vision and Image Understandings*, vol. 89, no. 2-3, pp. 114-141, Feb.-Mar. 2003.

[10] O. Javed, K. Shafique, Z. Rasheed, and M. Shah, "Modeling inter-camera space–time and appearance relationships for tracking across non-overlapping views," *Computer Vision and Image Understanding*, vol. 109, no. 2, pp. 146-162, Feb. 2008.