Title

Integrating Social Grouping for Multitarget Tracking Across Cameras in a CRF Model

Journal

IEEE Transactions on Circuits and Systems for Video Technology, 27(11)

ISSN

1051-8215

Authors

Chen, Xiaojing

Bhanu, Bir

Publication Date

2017-11-01

DOI

10.1109/tcsvt.2016.2565978

Peer reviewed

Integrating Social Grouping for Multitarget Tracking Across Cameras in a CRF Model

Xiaojing Chen and Bir Bhanu, *Fellow, IEEE*

***Abstract*— Tracking multiple targets across nonoverlapping cameras aims at estimating the trajectories of all targets, and maintaining their identity labels consistent while they move from one camera to another. Matching targets from different cameras can be very challenging, as there might be significant appearance variation and the blind area between cameras makes the target's motion less predictable. Unlike most of the existing methods that only focus on modeling the appearance and spatiotemporal cues for inter-camera tracking, this paper presents a novel online learning approach that considers integrating high-level contextual information into the tracking system. The tracking problem is formulated using an online learned conditional random field (CRF) model that minimizes a global energy cost. Besides low-level information, social grouping behavior is explored in order to maintain targets' identities as they move across cameras. In the proposed method, pairwise grouping behavior of targets is first learned within each camera. During inter-camera tracking, track associations that maintain single camera grouping consistencies are preferred. In addition, we introduce an iterative algorithm to find a good solution for the CRF model. Comparison experiments on several challenging real-world multicamera video sequences show that the proposed method is effective and outperforms the state-of-the-art approaches.**

***Index Terms*— Conditional random field (CRF) model, multitarget tracking, social grouping behavior.**

I. Introduction

With more and more surveillance cameras deployed at public places (e.g., airports, parking lots, and shopping malls) to monitor a large area, the demand for effective and automated surveillance and monitoring systems is rapidly growing [1]–[6]. Since using multiple cameras with overlapping field-of-views (FOVs) is not cost efficient in both the economical and computational aspects, cameras with nonoverlapping FOVs are widely used in real-world applications. Tracking multiple targets across nonoverlapping cameras is of great importance, as it is crucial for many industrial applications and high-level analysis, such as anomaly detection, crowd analysis, and activity detection and recognition.

Manuscript received October 29, 2015; revised February 23, 2016; accepted April 22, 2016. Date of publication May 10, 2016; date of current version November 8, 2017. This work was supported in part by the National Science Foundation under Grant 1330110 and in part by the Office of Naval Research under Grant N00014-12-1-1026. This paper was recommended by Associate Editor H. Wang.

X. Chen is with the Department of Computer Science, University of California at Riverside, Riverside, CA 92521 USA (e-mail: xchen010@ucr.edu).

B. Bhanu is with the Center for Research in Intelligent Systems, University of California at Riverside, Riverside, CA 92521 USA (e-mail: bhanu@cris.ucr.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2016.2565978

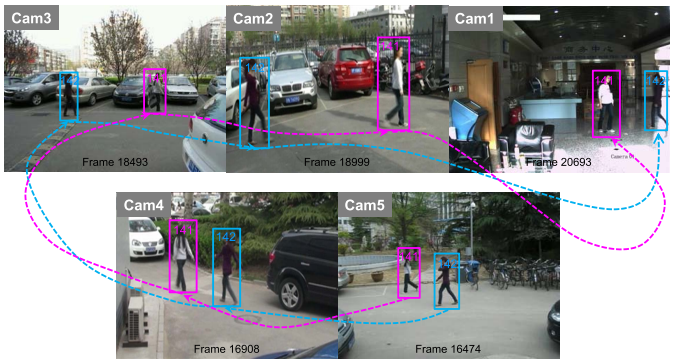


Fig. 1. Tracking results of our proposed model on Dataset4 [7]. Bounding boxes with the same color indicate the same person, and the dashed lines illustrate the trajectories generated by targets walking across different cameras.

Although there have been some improvements in this area, it remains a much lesser explored topic compared with single camera multitarget tracking.

The goal of multitarget tracking across nonoverlapping cameras is to automatically recover the trajectories of all targets and keep their identities consistent while they travel from one camera to another, as shown in Fig. 1. As compared with single camera tracking, where successive observations of the same target are likely to have a large similarity in appearance, across space and time [8], tracking across nonoverlapping cameras is a more challenging task due to the following factors.

1) *Significant Appearance Variation:* In multicamera tracking, the observations of the same target in different cameras often have significant differences, caused by illumination variation, pose change, and difference in sensor characteristics.
2) *Less Predictable Motion:* The open blind area between the FOVs of nonoverlapping cameras makes the motion prediction for each target less reliable. When a target leaves the FOV of one camera, it may enter the FOV of another camera, or exit from the region (under surveillance) in the blind area.

In most of the existing inter-camera multitarget tracking approaches, first, intra-camera tracking is carried out in each camera to produce tracks of different targets, and then, inter-camera tracking is conducted in the form of track association so that consistent labeling of each target across cameras can be achieved. To match tracks from different cameras, prior work mainly relies on the appearance and spatiotemporal cues. However, such low-level information is often unreliable especially for tracking in nonoverlapping cameras,

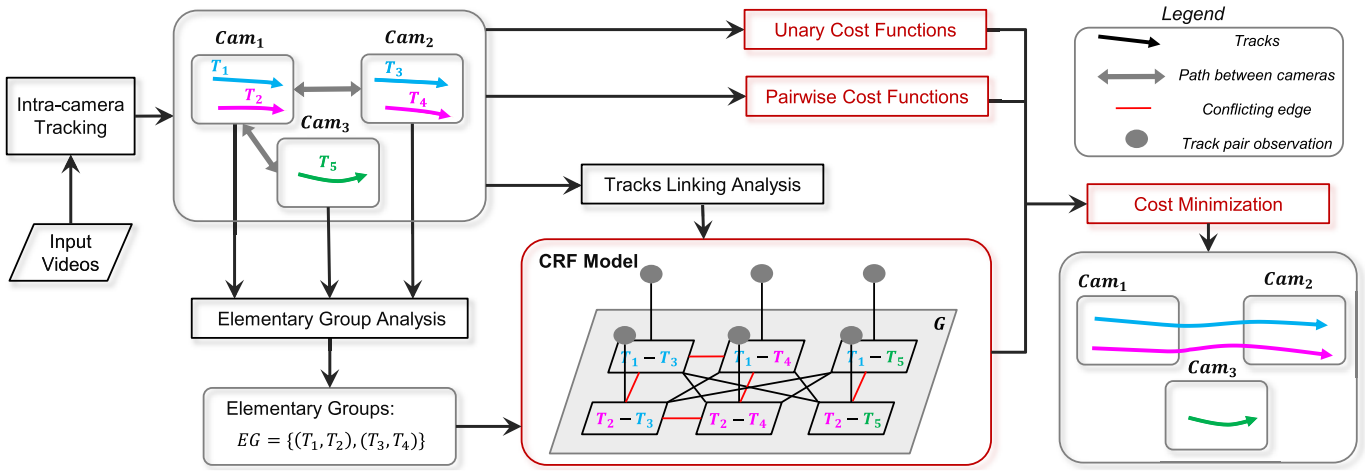


Fig. 2. Block diagram of our tracking system shown with a simple illustrative example. Blocks shown in red contain novel aspects of this paper. Tracks with the same color contain the same target. Best viewed in color. For the legends please see the box in the upper right-hand side.

as discussed earlier. In this paper, we consider integrating high-level contextual information, i.e., social grouping behavior, to mitigate ambiguities in inter-camera tracking.

Sociologists have found that up to 70% of people tend to walk in groups in a crowd for better group interaction [9], [10]. In addition, the leader–follower phenomenon generally exists in reality, which means pedestrians are likely to follow other individuals with the same destination either consciously or unconsciously to facilitate navigation [11]. Therefore, when two people are observed walking together in one camera for some time, it is very likely that these two people will appear together in a neighboring camera (see the example shown in Fig. 1). Based on the above observations, we propose an online learning approach for inter-camera tracking, which favors track associations that maintain group consistency. Note that we not only focus on groups that are formed by people who know each other, but are also interested in groups of individuals who have correlated movement.

We assume that the intra-camera tracking results for all the cameras are given, and the topology graph for cameras is known. To associate tracks from different cameras that contain the same person, an online learned conditional random field (CRF) model is used, as shown in Fig. 2. Track pairs that are linkable under certain spatiotemporal constraints form the nodes in the CRF model. Each node has a binary label (1 or 0) indicating whether the corresponding two tracks are linked or not in the final tracking result. A global appearance model is used to estimate the energy cost for each node. We use elementary groups [8] to analyze group status in each single camera. Two tracks form an elementary group if they have similar motion pattern and are temporally close to each other. Single camera grouping information is used to infer across camera grouping behavior. If two nodes in the CRF model contain at least one elementary group, an edge is created between them. Energy cost for each edge is estimated using the combination of both grouping and appearance information. For each track, we learn a target-specific appearance model online using AdaBoost. If two linked nodes not only have a high probability to maintain group consistency across cameras, but also have high appearance affinities according to target-specific appearance models, their corresponding edge will be assigned a small energy cost. Then, the tracking task is formulated as an energy minimization problem, i.e., to find the label assignment for the CRF graph that produces the smallest overall energy cost.

The rest of this paper is organized as follows. Section II discusses related work and presents contributions of this paper. The proposed CRF model and its corresponding approximation algorithm are described in Section III. Experiments are given in Section IV. Section V concludes this paper and provides possible directions for future work.

II. Related Work and Contributions

A. Related Work

Multitarget tracking across cameras has been an active topic in computer vision for many years, a recent comprehensive survey for this problem can be found in [5]. In general, methods for cross-camera multitarget tracking can be categorized into two groups according to the structure of camera networks: methods for overlapping FOVs and methods for nonoverlapping FOVs. Techniques used for tracking in these two groups have significant differences. For instance, tracking in a camera network with the overlapping FOVs normally require explicit camera calibration [12], [13], while it is not a necessity for tracking with nonoverlapping FOVs. Thus, publicly available tracking data sets are designed either for overlapping cameras or nonoverlapping cameras. In this paper, we focus on inter-camera tracking with nonoverlapping FOVs. This multitarget tracking is essentially a data association task, in which tracks from the same subject are to be matched. Due to the illumination and pose change across cameras, such data association is quite challenging.

Among various approaches for multitarget tracking, the appearance cue is commonly utilized. To tackle illumination change, brightness transfer functions (BTFs) have been exploited [14]–[16]. The BTFs model color changes between a pair of cameras through mapping functions. Variations of

BTFs include multivariate probability density function [15], joint brightness, tangent functions [17], and so on. Different BTFs are evaluated in [18], and the findings suggest that under certain conditions, such as during the entrance of a new subject, BTFs are prone to error. Besides BTFs, color correction models can also be used for tracking objects [19], [20]. In general, learning BTFs or color correction model requires large amount of training data and these models may not be robust against drastic illumination changes across different cameras.

In addition, spatiotemporal cue can be combined with appearance cue to improve the multitarget tracking performance. For example, Kuo *et al.* [21] learned a discriminative appearance model in a multiple instance learning framework, which can combine multiple descriptors and similarity measures effectively. This appearance model is used in conjunction with spatiotemporal information for improved tracking accuracy. Gao *et al.* [22] jointly exploits both the spatial and the temporal correlation from multiple tracking results, and achieves the state-of-the-art tracking performance. In particular, the temporal information is used to identify tracking consistency, and the spatial information helps to establish pairwise correlation among multiple tracking results. The work in [23] exploits spatiotemporal relationships among the targets to identify group merge and split events with time. It is designed to simultaneously track individuals and groups in a camera network, which is important for the problem of tracking in a cluttered scene. In addition, both spatiotemporal context and relative appearance context can be used jointly for inter-camera multitarget tracking. For example, in [24], the spatiotemporal cue supports sample collection for appearance model learning, and the relative appearance context helps disambiguate people in proximity. An inter-camera transfer model, including both the spatiotemporal and appearance cues, is proposed in [25]. In particular, the spatiotemporal model is learned using an unsupervised topology recovering approach, and the appearance model is learned by modeling color changes across cameras.

Another recently popular research topic, person reidentification, is closely related to inter-camera multitarget tracking. Both problems aim to match the observations of the same people across nonoverlapping cameras. However, in most person reidentification work, only a single or multiple snapshots of people are to be matched. Therefore, contextual information is often not available for person reidentification problem. On the other hand, in an inter-camera tracking problem, each person is presented by a track, which is a string of detections extracted from consecutive frames. In order to handle the large intra-class variation in person reidentification, the robust appearance models have been studied [26]–[29]. Another way is to learn specialized distance metrics or feature transformations [30]–[34]. For training purpose, a training set with corresponding detection pairs, which share similar imaging conditions as the testing samples, is required.

While most of the previous work (e.g., [25], [35]) only considers pairwise relationships using global optimization techniques, such as Hungarian algorithm, we employ CRF to simultaneously model both the pairwise and higher order relationships for track association. The CRF model has been adopted in several single camera multitarget tracking papers. Yang and Nevatia [36] use CRF to learn the dependencies and distinguish the difficult pairs of tracklets in single camera tracking. Heili *et al.* [37] formulate the tracking problem using a CRF framework with pairwise similarity and dissimilarity hypotheses. Compared with person reidentification, in which only images of the subjects are matched, our framework is a dynamic system, meaning that the track association is executed to cover both the spatial and temporal spans. Such a system is more desirable for tracking and monitoring in practical applications.

B. Contributions of This Paper

As compared with the state of the art described earlier in this paper, the contributions of this paper are as follows.

1) A novel CRF framework that combines social grouping behavior with traditionally used appearance and spatiotemporal cues for robust multitarget tracking across nonoverlapping cameras.
2) An online learning approach for modeling the unary and pairwise energy costs in the CRF model. The proposed approach does not require a large training set with known correspondence between samples, and it is able to adapt environmental changes.
3) An effective approximation algorithm for the CRF model that produces good tracking results with low energy cost.
4) Evaluation on four challenging real-world surveillance video sequences is used to validate the effectiveness of the proposed method.

III. Technical Approach

A. CRF Model for Inter-Camera Tracking

In this section, we formulate inter-camera tracking as an inference problem using the CRF framework. An outline of the proposed tracking system is shown in Fig. 2.

Given a set of tracks $T = \{T_1, T_2, \ldots, T_N\}$, which is the result of intra-camera tracking of M nonoverlapping cameras $\mathrm{Cam}_1, \mathrm{Cam}_2, \ldots, \mathrm{Cam}_M$. Each track T_i is a string of detections that correspond to the same person and they are extracted from a set of continuous frames. The time interval for T_i is denoted as $[t_i^{\mathrm{begin}}, t_i^{\mathrm{end}}]$, and its corresponding camera is $\mathrm{Cam}(T_i)$. The task of inter-camera multitarget tracking is to associate tracks from different cameras that contain the same person under certain spatiotemporal constraints. Since the CRF framework is capable of encoding relationship between observations, it is especially suitable for capturing and modeling contextual information in the scene.

We create a CRF graph $G = \{V, E\}$. Each vertex $v_i = (T_i^1, T_i^2)$ in V represents a linkable pair of tracks, assuming T_i^1 starts before T_i^2, and each edge $e_j = (v_j^1, v_j^2)$ in E indicates that the connected two vertices are correlated (detailed explanations for CRF graph creation is presented in Section III-B). Let $L = \{l_1, l_2, \ldots, l_m\}$ be a set of binary labels for all vertices, i.e., all possible track associations, with $l_i = 1$ indicating T_i^1 is associated with T_i^2 in the final tracking result,

and $l_i = 0$ represents the opposite. During tracking, our goal is to find the label configuration L^* that maximizes the overall linking probability, given the value of T. Mathematically, the inter-camera tracking problem can be defined by the following optimization equation:

$$L^* = \arg\max_L P(L|T) = \arg\min_L \frac{1}{Z}\exp(-\Psi(L|T)) \quad (1)$$

where Z is a normalization factor that does not depend on L, and $\Psi(\cdot)$ is a potential/cost function. We assume that the joint distributions of more than two associations have no contributions to the conditional probability $P(L|T)$, then

$$\begin{aligned} L^* &= \arg\min_L \Psi(L|T) \\ &= \arg\min_L \left(\sum_i U(l_i|T) + \sum_{i,j} B(l_i, l_j|T) \right) \end{aligned} \quad (2)$$

where $U(l_i|T)$ and $B(l_i, l_j|T)$ are the unary and pairwise energy functions and correspond to the node and edge costs in the CRF graph, respectively. Learning of the unary and pairwise costs are described in Sections III-C and III-D.

For efficiency, track association is not applied on the entire videos. Instead, a predefined time sliding window is used, and a CRF model is learned online for each sliding window. Moreover, in order to prevent impractical associations, a valid label set L needs to follow certain constraints. Let L^1 be the set of all labels that are assigned to 1, namely, $L^1 = \{l_i = 1\}\ \forall l_i \in L$. Similarly, L^0 corresponds to the set of labels assigned with 0. For a label l_k, with its corresponding vertex denoted as $v_k = \{T_k^1, T_k^2\}$, we use $C(l_k)$ to represent the set of its conflicting labels. A label l_x is conflicting to l_k, if its corresponding vertex $v_x = \{T_x^1, T_x^2\}$ falls into one of the following patterns: 1) $T_x^1 = T_k^1$ and $T_x^2 \neq T_k^2$ and 2) $T_x^2 = T_k^2$ and $T_x^1 \neq T_k^1$. Then, L is a valid label set, if

$$\forall l_k \in L^1, \quad C(l_k) \subset L^0. \quad (3)$$

This constraint implies that each track can be associated with and associated by only one other track.

B. CRF Graph Creation

In the CRF graph, each vertex represents a pair of linkable tracks. Track T_i can be associated with T_j if they satisfy the following spatiotemporal constraints.

1) *Spatial Constraints:* First, T_i and T_j are captured in different cameras, namely, $\text{Cam}(T_i) \neq \text{Cam}(T_j)$. Second, according to the camera topology graph, linking T_i and T_j forms a feasible path allowing people to walk from $\text{Cam}(T_i)$ to $\text{Cam}(T_j)$ without entering the FOV of any other cameras.
2) *Temporal Constraints:* T_i starts before T_j. Let $\text{Gap}_{ij} = t_j^{\text{begin}} - t_i^{\text{end}}$ be the time gap between these two tracks, then $0 < \text{Gap}_{ij} < \text{GAP}$ should hold, where GAP is a threshold for maximum time gap between any two linkable tracks.

The spatial constraints enable us to focus only on inter-camera tracking, as well as eliminate those practically infeasible track associations. The temporal constraints prevent us from linking track pairs with time overlap, as one individual cannot appear at two different places at the same time. The threshold GAP avoids track pairs outside the time sliding window to be considered.

Given a set of tracks, the linkability of any two tracks is evaluated according to the above spatiotemporal constraints. A set of vertices V is created, and each vertex in V denotes a pair of linkable tracks as

$$\begin{aligned} V = \{ v_i = (T_i^1, T_i^2) \} \\ \text{s.t. } T_i^1 \text{ can be linked to } T_i^2. \end{aligned} \quad (4)$$

In order to build edges between the vertices in the CRF graph, we first find elementary groups in each single camera. Elementary group is a flexible structure for within-camera grouping analysis [8]. An elementary group is a group including only two people, and they move with similar motion pattern and are temporally close to each other. Because the number of groups and the sizes of groups in the scene are unknown and may change over time, learning the complete group structure directly is quite challenging. Elementary group provides a simple but effective way for inferring useful group information, since a group of any size can be presented by a set of elementary groups. Note that the elementary group analysis is carried out in an online mode.

In a single camera, track T_i forms an elementary group with T_j if they have the following properties: 1) T_i and T_j co-exist for at least t s (t is set to 2 in our experiments) and 2) the angle between the velocities of T_i and T_j is smaller than 45°. The first constraint guarantees that the two tracks in an elementary group are temporally close to each other. As we assume there is only a small variation in the walking speed of most pedestrians, two targets are considered as dynamically correlated if they walk toward approximately the same direction. We consider only the moving direction of a target and do not use its walking speed during elementary group learning. Because the velocity of each target is calculated based on its corresponding detections on the image plane using a Kalman filter. The computed velocity is not as precise as the one obtained by projecting locations of a target into the $3D$ scene coordinates. However, camera parameters are required for computing such projection, and not every surveillance video sequence comes with this information.

Unlike [8], we relax the elementary group criterion by removing the spatially close constraint. The groups we are interested in not only refer to acquaintances who are walking side by side, but also include people who do not know each other but have similar motion pattern, such as one person is following the other toward the same destination. In the second situation, the distance between the two pedestrians may not be very close. Thus, interpersonal distance is ignored when forming elementary groups, so that more leader–follower instances can be detected. Moreover, for pedestrians that are too far away, they will not appear together for a long time in the same scene, as one would exit the scene much earlier

than the other. We assume that if two people are observed appearing together and have similar motion pattern in one camera, then they are likely to reappear together in a neighboring camera. Thus, we can improve track association across cameras with elementary grouping information in each single camera.

Let $EG = \{g_i = (T_{g_i}^1, T_{g_i}^2)\}$ be the set of elementary groups found in all cameras. An edge is created for two vertices $v_i = (T_i^1, T_i^2)$ and $v_j = (T_j^1, T_j^2)$, if at least one elementary group can be formed by the four involved tracks. Mathematically, we define a set of edges E for the CRF graph as

$$E = \{(v_i, v_j)\} \quad \forall v_i,\ v_j \in V \\ \text{s.t. } \left(T_i^1, T_j^1\right) \in EG \text{ or } \left(T_i^2, T_j^2\right) \in EG. \tag{5}$$

Moreover, edges are divided into conflicting ones and non-conflicting ones. A conflicting edge means that the connected two vertices cannot be assigned with label 1 at the same time, in order to guarantee a valid label set. Note that the edges are created between vertices containing targets with the same motion direction, e.g., from Cam_1 to Cam_2. During tracking, the set of track pairs that maintains the overall group consistency are more likely to be associated. In the example shown in Fig. 2, two elementary groups (T_1, T_2) and (T_3, T_4) are found based on all the input tracks. Therefore, if we know T_1 and T_3 have a high probability to be associated, then the probability for linking T_2 and T_4 should be increased, as the same group of people are likely to reappear together in a neighboring camera. Besides overall group consistency, the associated tracks should also keep appearance consistency according to the online learned target-specific appearance models. Both the group and the appearance consistency are estimated using the online learned pairwise costs (see Section III-D).

C. Unary Energy Functions

Unary energy functions in (2) evaluate the energy cost for linking two tracks. The cost is defined as the negative log-likelihood of two tracks being the same target according to a global appearance model $P_{\text{app}_1}(\cdot)$

$$U(l_i = 1|T) = -\ln P_{\text{app}_1}\left(T_i^1, T_i^2|T\right). \tag{6}$$

1) Track Division: In a track, detections from adjacent frames often have high appearance similarity. In order to reduce redundancy and create concise and robust representation, for each track we combine visually similar detections into a subtrack and consider each subtrack as an appearance instance of a target, as used in [35].

More specifically, given a track, its first detection is used as a reference detection for its first subtrack. Following detections that have high appearance similarity ($\geq$0.9) compared with the reference detection are included into the first subtrack. When the similarity of a detection to the reference detection is below 0.9, this detection is considered as the reference detection for the next subtrack. The process continues until we reach the end of the track. In addition, we set the maximal length of a subtrack to 20 frames (about 1 s) to ensure that there is no large pose variation for detections contained in the same subtrack.

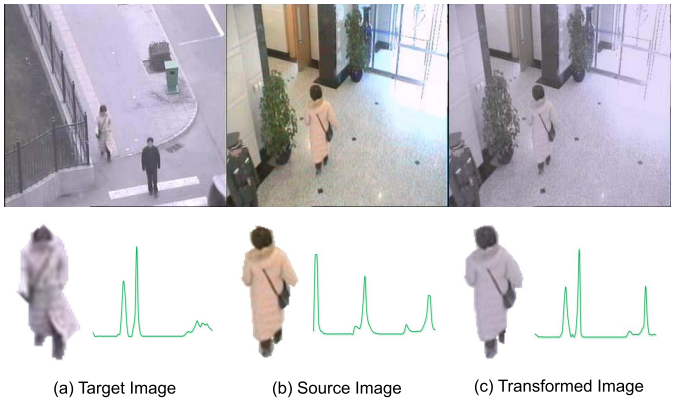


Fig. 3. Example of applying the color transfer method on images obtained by two different cameras. (a) $\text{Cam}_{\text{target}}$ (outdoor). (b) $\text{Cam}_{\text{source}}$ (indoor). The full image frames captured by each camera are shown in the first row. The person appears in both cameras and its corresponding HSV color histograms are presented in the second row. It is obvious that the person in (c) transformed image is more like the person in the target image based on HSV color histograms.

2) Color Transfer: In order to compute appearance similarity of tracks from different cameras, we first need to handle appearance variance across cameras. In this paper, we adopted the color transfer method proposed in [25] and [38] as a preprocessing step to normalize color between different cameras. Given two images, the color transfer method achieves color normalization by imposing the color characteristics of one image (target image) onto the other (source image), as shown in Fig. 3. In our experiments, the first full image from one camera is used as the target image, and the images from other cameras are considered as source images.

As correlations exist among the three different color channels of the RGB color space [38], to change the color of one pixel, the values of this pixel in all channels must be modified. Such correlations are undesirable for color transfer. Therefore, images are transferred from the original RGB color space to the $l\alpha\beta$ color space, where there is little correlation between different color channels. Then, the target image is transformed according to the color characteristics exacted from the source image, as

$$l^* = \frac{\sigma_t^l}{\sigma_s^l}\left(l_s - m_s^l\right) + m_t^l \\ \alpha^* = \frac{\sigma_t^\alpha}{\sigma_s^\alpha}\left(\alpha_s - m_s^\alpha\right) + m_t^\alpha \\ \beta^* = \frac{\sigma_t^\beta}{\sigma_s^\beta}\left(\beta_s - m_s^\beta\right) + m_t^\beta \tag{7}$$

where l, α, and β represent the pixel value in a corresponding color channel, m and σ denote the mean and standard deviation of one image. The target and source images are indexed by subscript t and s, respectively. $[l^*, \alpha^*, \beta^*]$ is the representation of the transformed image in the $l\alpha\beta$ color space. After color transformation, the transformed image is converted back to the RGB color space from the $l\alpha\beta$ color space.

Given two tracks T_i and T_j with $\text{Cam}(T_i) \neq \text{Cam}(T_j)$, HSV color histograms are extracted from each detection. The average of HSV color histograms from the same subtrack is

regarded as one appearance descriptor for the target contained in the track. W subtracks are randomly selected from each track, and their average similarity is used as the similarity for T_i and T_j. Mathematically, the global appearance model for T_i and T_j is defined as

$$P_{\text{app}_1}(T_i, T_j|T) = \frac{1}{W}\sum_{n=1}^{W} \text{BC}\left(d_n^i, d_n^j\right) \tag{8}$$

where d_n^i is the nth randomly selected subtrack in track T_i. BC$(\cdot)$ is the Bhattacharyya coefficient [39], it is used as a measure for the appearance similarity of two subtracks.

D. Pairwise Energy Functions

The pairwise energy functions are formulated according to the global grouping cues and the target-specific appearance cues, as defined in

$$B(l_i, l_j|T) = -\ln(P_{\text{group}}(l_i, l_j|T) \times P_{\text{app}_2}(l_i, l_j|T)) \tag{9}$$

where P_{group} is the probability of maintaining group consistency for a specific assignment of (l_i, l_j), and P_{app_2} is the probability of keeping appearance consistency based on the value of l_i and l_j. Details for P_{group} and P_{app_2} are presented in the following parts.

1) Group Consistency: According to the observation that two people walking together for a certain time in one camera are likely to reappear together in a neighboring camera, given the labels of two connected vertices in the graph, we can infer its probability of maintaining group consistency.

Let $v_i = (T_i^1, T_i^2)$ and $v_j = (T_j^1, T_j^2)$ be two possible track associations, without knowing the edge configuration of the graph, the probability of maintaining group consistency for a specific label assignment of (l_i, l_j) is $1/C$, where C is the number of all possible values for (l_i, l_j). Assuming we know v_i is connected to v_j in the graph, which indicates that $\text{Cam}(T_i^1) = \text{Cam}(T_j^1)$, $\text{Cam}(T_i^2) = \text{Cam}(T_j^2)$. If both (T_i^1, T_j^1) and (T_i^2, T_j^2) are elementary group, then assigning (l_i, l_j) to $(1, 1)$ maintains the group consistency. For instance, in the example shown in Fig. 2, as (T_1, T_2) and (T_3, T_4) are both elementary groups in Cam_1 and Cam_2, then assigning $(1, 1)$ to vertices (T_1, T_3) and (T_2, T_4) keeps the group consistency compared with the other alternatives [i.e., $(1, 0)$, $(0, 1)$, and $(0, 0)$]. Based on the above analysis, a group consistency score S_{group} for each edge is defined below

$$S_{\text{group}}(l_i, l_j|T) = \begin{cases} 1, & \text{if } \left(T_i^1, T_j^1\right) \in EG, \ \left(T_i^2, T_j^2\right) \in EG \\ & \text{and } l_i = l_j = 1 \\ \dfrac{1}{C}, & \text{otherwise.} \end{cases} \tag{10}$$

To define P_{group}, we normalize S_{group} to have a summation equal to one to fit the requirement of a probability, as

$$P_{\text{group}}(l_i, l_j|T) = \frac{S_{\text{group}}(l_i, l_j|T)}{\sum_{l_i, l_j} S_{\text{group}}(li, lj|T)}. \tag{11}$$

Note that if (v_i, v_j) is a nonconflicting edge, $C = 4$, as there are four possibilities, i.e., $(1, 1)$, $(1, 0)$, $(0, 1)$, $(0, 0)$, for the label assignment of (l_i, l_j). But if (v_i, v_j) is a conflicting edge, indicating l_i and l_j cannot have label 1 at the same time, then $C = 3$ for such cases.

2) Local Appearance Consistency: It is obvious that from group consistency alone we cannot obtain sufficient information to make confident track association decisions. Therefore, we integrate local appearance consistency into the pairwise energy functions. An edge possesses local appearance consistency if the label given to each related vertex is in accordance with appearance similarity/dissimilarity of the corresponding track pair.

Mathematically, given an edge (v_i, v_j), where v_i contains track pair (T_i^1, T_i^2) and v_j includes (T_j^1, T_j^2). Let App_{ik} be a discriminative appearance model learned for track T_i^k, which produces high similarity for a track that contain similar target as T_i^k, and gives low similarity otherwise. Then, we define P_{app_2} as

$$\begin{aligned} P_{\text{app}_2}(l_i = 1, l_j = 1|T) &= P(l_i = 1)P(l_j = 1) \\ P_{\text{app}_2}(l_i = 1, l_j = 0|T) &= P(l_i = 1)(1 - P(l_j = 1)) \\ P_{\text{app}_2}(l_i = 0, l_j = 1|T) &= (1 - P(l_i = 1))P(l_j = 1) \\ P_{\text{app}_2}(l_i = 0, l_j = 0|T) &= (1 - P(l_i = 1))(1 - P(l_j = 1)) \end{aligned} \tag{12}$$

where P is the probability of two tracks that contain the same person based on the discriminative appearance model App. It is defined as $P(l_i = 1) = 0.5 \times (\text{App}_{i1}(T_i^2) + \text{App}_{i2}(T_i^1))$.

The discriminative appearance model for each track is learned online using AdaBoost. We capture the appearance information of each target using various features, such as HSV color histograms [40], local binary pattern (LBP) [41], histogram of gradient (HOG) [42], and color names [43]. Each feature descriptor is computed at different local patches defined on a detection, as shown in Fig. 4. We resize each detection to 63×27, and extract the target contained therein using background subtraction. Local patches are defined at different locations with various scales to increase the descriptive ability. Features of the same type in one subtrack are averaged to construct a concise representation for the target. In general, one track may contain several subtracks, and there are in total $15 \times 4 = 60$ features for each subtrack.

Given two subtracks t_a and t_b, comparing each of the 60 appearance feature descriptors produces one appearance similarity. A concatenation of the 60 similarities scores forms a feature vector $f(t_a, t_b)$. In our experiments, different methods

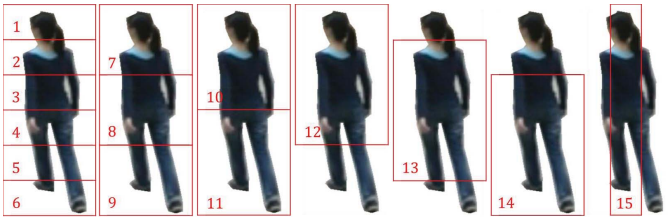


Fig. 4. Local patches (indexed from 1 to 15) with various scales are defined at different locations of a detection. Patches 1–6 have the same size and are served as basic patches, patches 7–14 are different combinations of basic patches, and patch 15 captures the middle third region of a detection.

are used to measure the similarity between different types of features. Bhattacharyya coefficient [39] is used for color histograms and HOG features, χ^2 distance is used for LBP features, and cosine similarity is used for color names.

AdaBoost adaptively learns a strong classifier using a number of weak classifiers that minimizes the overall classification error. The generated strong classifier is a linear combination of weak classifiers, and the weight for each selected weak classifier indicates its importance. In our target-specific appearance model, the similarity computed from each feature is used in a weak classifier, and the learned appearance model is formulated as

$$H(f(t_a, t_b)) = \sum_{t=1}^{T} \alpha_t h_t(f(t_a, t_b)) \tag{13}$$

where T is the number of total iterations, α_t is the weighting parameter assigned during the learning process, and $h_t(f(t_a, t_b))$ is a weak classifier based on one of the features extracted from subtracks t_a and t_b.

In order to learn the discriminative appearance model online for each target, we collect training samples during track association. Given a track T_x, a pair of subtracks can form a positive training sample if they are two different subtracks in T_x. A negative sample can be generated by two subtracks if one of them is from T_x, and the other is from another track that has time overlap with T_x. Therefore, a positive sample consists of feature similarities of the same target, while in a negative sample the feature similarities are calculated from two different targets.

Once the discriminative appearance model is learned for a target, we can compute the appearance similarity between this target and the other targets using

$$\mathrm{App}_{i1}\left(T_i^2\right) = \sum_{w=1}^{W} H_{i1}\left(f\left(t_{i1}^{w}, t_{i2}^{w}\right)\right) \tag{14}$$

where App_{i1} is the target-specific appearance model learned for track T_i^1, and it is used to compute the similarity between T_i^1 and T_i^2. We randomly select W subtrack pairs from both tracks, and use the average of their similarity for the similarity of the track pair.

Note that both the global appearance model P_{app_1} in the unary energy function and the local appearance model P_{app_2} in the pairwise energy function are defined to compute the visual similarity of two linkable tracks. However, the local appearance model in the pairwise energy function only applies to the vertices that have edge in the CRF graph, while all the vertices in the CRF graph are required to compute the global appearance affinity using the unary energy function. The local appearance model is generated by online AdaBoost learning, which has higher computational cost as compared with the global appearance model. Therefore, to improve the efficiency, local appearance model is not integrated in the unary energy function.

In the proposed tracking method, appearance affinities of all linkable track pairs are computed. The association of two tracks indicates that they are visually similar. Keeping the appearance consistency of each member in the group implies that the group-level appearance consistency is also maintained.

E. Energy Minimization Algorithm

We formulated the across-camera multitarget tracking task as an energy minimization problem using CRF model, as shown in (2). Since the proposed CRF model does not follow the submodularity principle (see the Appendix), we cannot obtain exact inference using the global graph cut optimization techniques [44]. Moreover, traditional approximation approaches for CRF, such as loopy belief propagation (LBP) and alpha expansion, cannot be directly applied for our problem, as solutions produced by these methods may not satisfy the constraint for a valid label set [see (3)]. Therefore, we developed an iterative approximation algorithm to find a good labeling solution.

More precisely, we first obtain an initial labeling of all vertices using only Hungarian algorithm with unary costs, similar to [35]. As Hungarian algorithm allows only one assignment for each participant, this ensures the initial label set to be a valid one. Then, vertices assigned with label 1, i.e., the selected track associations, are sorted in ascending order according to their unary costs. Next, for each label 1 vertex, we find all edges that are connected to the current vertex. For each of these edges, all other label configurations are considered, and the one with the minimal graph energy cost is selected. Note that, for a conflicting edge, there are only three labeling possibilities: $(1, 0)$, $(0, 1)$, and $(0, 0)$. If the chosen label configuration generates an energy cost lower than the current one, we update the label set with the change. In order to maintain the constraints for a valid label set, each time when the label of a vertex changes from 0 to 1, we check if the constraint in (3) is violated. When violations exist, the new update is preferred.

A summary of the energy minimization algorithm is provided in Algorithm 1.

Our proposed energy minimization algorithm finds the label set in a greedy fashion, thus, it may lead to a local optimal solution. However, a better solution, i.e., a label set with lower energy cost, is achieved after each iteration. Therefore, it ensures that we achieve better tracking results than using unary costs only.

IV. Experiments

To validate the effectiveness of the proposed tracking approach, it is compared with several baseline methods and the state-of-the-art approaches. We carried out experiments on four different sets of data that are publicly available.

A. Data Sets

Although multitarget tracking in surveillance cameras has been studied for several years, there are fewer public data sets available for real-world multicamera tracking as compared with single camera tracking. In this paper, we use the NLPR_MCT data set [7] to evaluate the performance of the proposed method. The NLPR_MCT data set has both

Algorithm 1 Algorithm for Finding Labels With Low Energy Cost

Input: Tracklet set $T = \{T_1, .., T_n\}$; CRF graph $G = \{V, E\}$
Output: A label set L

Use Hungarian algorithm to find an initial label set L with the lowest unary energy cost, and evaluate current graph energy cost Ψ in Eq. 2.
Sort label 1 vertices according to their unary costs as $\{v_1, ..., v_m\}$
for $i = 1, ..., m$ **do**
  Find a set E_i including all edges connecting to v_i
  Set updated graph energy cost $\Psi^{'} = +\infty$
  for all $e = (v_i, v_x) \in E_i$ **do**
    Change labels of (v_i, v_x) to a untested possibility,
    maintain constraints for a valid label set,
    evaluate the new graph energy cost Ψ_{new}
    if $\Psi_{new} < \Psi^{'}$ **then**
      $\Psi^{'} = \Psi_{new}$
  if $\Psi^{'} < \Psi$ **then**
    $\Psi = \Psi^{'}$
    Update L with the change

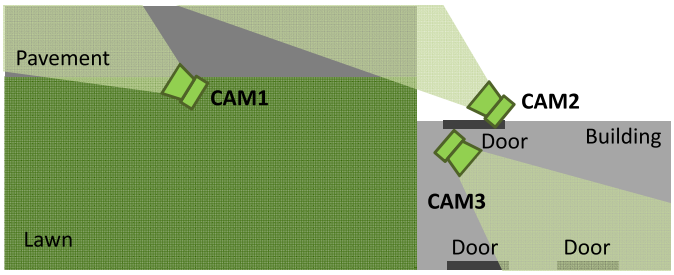


Fig. 5. Camera topology for Dataset1 and Dataset2. Cam_1 and Cam_2 are outdoor cameras, and Cam_3 is an indoor camera.

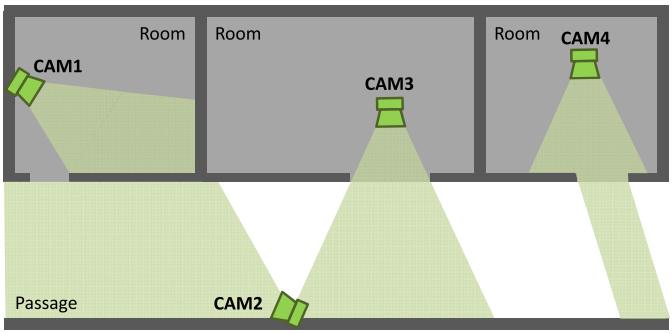


Fig. 6. Camera topology for Dataset3. Cam_1–Cam_4 are all indoor cameras.

the outdoor and indoor scenarios. In addition, there exists illumination variations across cameras, which makes it a very challenging data set for multitarget tracking.

Four different subdata sets are contained in the NLPR_MCT data set, and each of them corresponds to a nonoverlapping multicamera network. Dataset1 and Dataset2 have the same camera setting, including three cameras (two outdoor and one indoor), as shown in Fig. 5. Dataset3 contains four videos that are captured by four indoor cameras, the topology of these cameras is shown in Fig. 6. The nonoverlapping

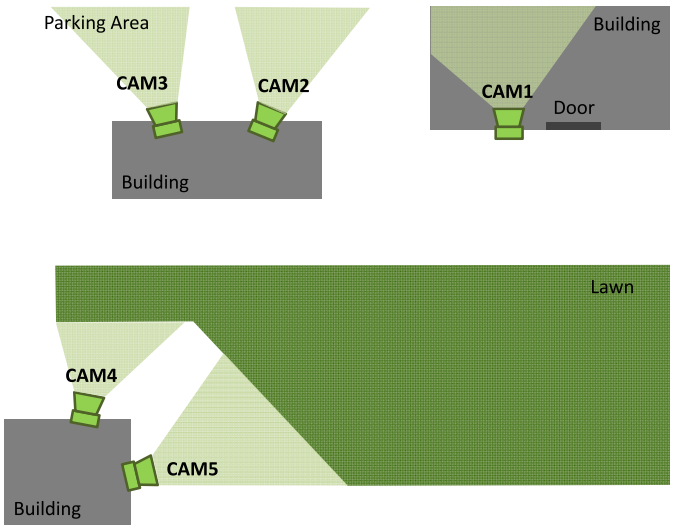


Fig. 7. Camera topology for Dataset4. Cam_1 is an indoor camera and Cam_2–Cam_5 are all outdoor cameras.

TABLE I
SPECIFICS FOR EACH SUBDATA SET IN THE NLPR_MCT DATA SET [7]

	Dataset1	Dataset2	Dataset3	Dataset4
# of Cameras	3	3	4	5
Resolution	320 × 240	320 × 240	320 × 240	320 × 240
Duration	20min	20min	3.5min	24min
# of Targets	235	255	14	49
Frame Rate	20fps	20fps	25fps	25fps

camera network of Dataset4 consists of four outdoor and one indoor cameras, the topology of these cameras is shown in Fig. 7. More specifics for each subdata set are listed in Table I.

It is clear that the quality of input tracks, i.e., within-camera tracking results, will greatly affect the performance of multitarget tracking across cameras. In order to have a fair comparison on the cross-camera tracking ability, first, we use the same input tracks for all the tested methods in our experiments. The input tracks are the single camera tracking ground truth provided in the NLPR_MCT data set. Then, the estimated single camera tracking results are used to validate the robustness of the proposed tracking method against imperfect input.

B. Evaluation Metrics

As has been noticed in multitarget tracking in a single camera that it is very difficult to have a direct quantitative comparison of different tracking approaches due to the lack of a standardized benchmark [46]. The same issue persists in multitarget tracking across cameras. Inspired by the widely used CLEAR MOT metrics [47] for single camera multiobject tracking, the NLPR_MCT data set provides an evaluation metric, multicamera tracking accuracy (MCTA), which is a single scalar metric that combines the detection accuracy (detection), the single camera tracking accuracy ($\text{tracking}^{\text{SCT}}$), and the inter-camera tracking accuracy ($\text{tracking}^{\text{ICT}}$). The definition of

Fig. 8. Visual comparison of our model (the first row) and Baseline1 (the second row) on Dataset1. It is observed that Baseline1 mistakenly identifies a new target in Camera 3 (the one pointed by arrow) as target 3, while our model avoid this error by maintaining the group consistency between targets 3 and 4. Bounding box with the same color indicates the same target. Best viewed in color.

MCTA is given below

$$\begin{aligned}\text{MCTA} &= \text{Detection} \times \text{Tracking}^{\text{SCT}} \times \text{Tracking}^{\text{ICT}} \\ &= \left(\frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}\right)\left(1 - \frac{\sum_t \text{mme}_t^s}{\sum_t \text{tp}_t^s}\right) \\ &\quad \times \left(1 - \frac{\sum_t \text{mme}_t^c}{\sum_t \text{tp}_t^c}\right) \end{aligned} \tag{15}$$

where precision and recall reflect the performance of the object detector, mme_t^s is the number of mismatches (i.e., ID-switches) for time t in a single camera, and mme_t^c is the number of mismatches for time t across different cameras, tp_t^s and tp_t^c are the number of true positives for time t within camera and across cameras, respectively. Note that, according to the defined criteria, when a new target first enters the scene, it produces a new cross-camera true positive instead of a within-camera true positive.

The MCTA metric ranges from 0 to 1, a higher value indicates a better tracking performance. In order to focus on the ability of across-camera multitarget tracking, we first use single camera tracking ground truth as input tracks. Therefore, the first two terms in (15), i.e., Detection and $\text{Tracking}^{\text{SCT}}$, both have value 1. The cross-camera tracking performance is only affected by mme^c, the number of mismatches across cameras.

C. Experimental Results

In this evaluation, our goal is to link tracks in different camera views that contain the same target under certain spatiotemporal constraints. The number of cross-camera true positives in each subdata set is shown in Table II. We introduce two baseline models for comparison.

TABLE II
NUMBER OF SINGLE CAMERA TRUE POSITIVES (TP_s) AND CROSS-CAMERA TRUE POSITIVES (TP_c) IN EACH SUBDATA SET

	Dataset1	Dataset2	Dataset3	Dataset4
TP_s	71853	88419	18187	42615
TP_c	334	408	152	256

1) *Baseline1:* Use only Hungarian algorithm with global appearance model, no grouping information is used.
2) *Baseline2:* Our proposed CRF model without the local appearance consistency in (9).

A quantitative comparison of our proposed model and the baseline models using single camera tracking ground truth as input is shown in Table III. It is observed that our proposed model significantly improves the tracking performance on all subdata sets compared with Baseline1. For Dataset1 and Dataset2, our model increases MCTA by almost 0.3. For Dataset3, the improvement with respect to MCTA is 0.25. The largest improvement is achieved in Dataset4, where the MCTA improves by 0.46 when our proposed model is used. Therefore, it is validated that by integrating social grouping information we can achieve a better tracking performance, since high-level context provides other useful information that is not included in low-level features. A visual comparison of our model and Baseline1 on Dataset1 is shown in Fig. 8.

In Baseline2, only group consistency is taken into account for edge cost calculation in the CRF graph. The tracking

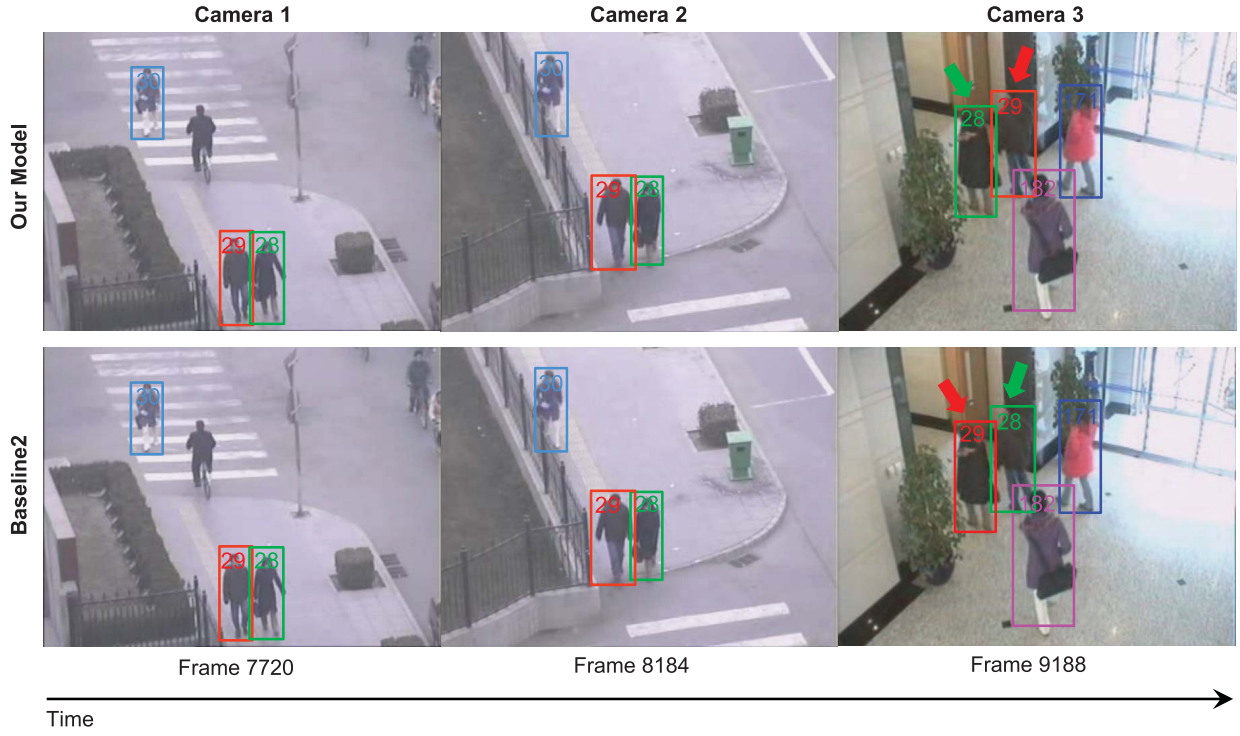


Fig. 9. Visual comparison of our model (the first row) and Baseline2 (the second row) on Dataset2. In the result of our model, target 28 and 29 are correctly tracked in all cameras. But their IDs are switched in Camera 3 in the result of Baseline2, due to the lack of local appearance consistency. Bounding box with the same color indicates the same target. Best viewed in color.

TABLE III
COMPARISON OF CROSS-CAMERA TRACKING RESULTS ON THE NLPR_MCT DATA SET WITH SINGLE CAMERA TRACKING GROUND TRUTH AS INPUT

Method	Dataset1		Dataset2		Dataset3		Dataset4	
	mme^c	MCTA	mme^c	MCTA	mme^c	MCTA	mme^c	MCTA
Baseline1	156	0.5329	197	0.5172	89	0.4145	150	0.4141
Baseline2	91	0.7275	102	0.7500	62	0.5921	118	0.5391
This paper	54	0.8383	81	0.8015	51	0.6645	70	0.7266
USC-Vision [24]	27	0.9152	34	0.9132	70	0.5163	72	0.7052
Hfutdspmct [45]	86	0.7425	141	0.6544	40	0.7368	155	0.3945
CRIPAC-MCT [45]	113	0.6617	167	0.5907	44	0.7105	110	0.5703

performances of Baseline2 on all subdata sets are better than that of Baseline1, which further validates the effectiveness of grouping information for track association. Comparison between our proposed model and Baseline2 indicates that local appearance consistency plays an important role in eliminating incorrect track association, as it requires that the linked track pair should not only have a high appearance similarity in the global appearance model but also be visually similar according to the local appearance model. A visual comparison of our model and Baseline2 on Dataset2 is shown in Fig. 9. In Fig. 10, we provide the tracking results of our model on Dataset3. More tracking results on Dataset4 using the proposed method are shown in Figs. 1 and 11.

In addition, the proposed CRF model is compared with other methods for tracking in multiple nonoverlapping cameras. These methods are reported in the multicamera object tracking (MCT) challenge [45] held at ECCV 2014 visual surveillance and reidentification workshop. We select the top three methods for comparison, their corresponding tracking performances on each subdata set with single camera ground truth as input are shown in Table III, with USC-Vision [24] being rank 1, Hfutdspmct being rank 2, and CRIPAC-MCT being rank 3.

The proposed tracking method takes advantage of social grouping information to produce robust track association. Thus, the more occurrences of across camera grouping, the more benefit can be gained. By analyzing the videos in the four subdata sets, we found that the number of times that two pedestrians walking together in one camera for a while and then reappear together in a neighboring camera are 32, 79, 14, and 57 for Dataset1–Dataset4, respectively. Since there are 334, 408, 152, and 256 cross-camera true positive track associations in the four subdata sets, respectively (see Table II), the percentages of true positive track associations that involve across camera grouping are 19.2%, 38.7%, 18.4%, and 44.5% for Dataset1–Dataset4. According to the results shown in Table III, our proposed model takes advantage of the adequate grouping information contained in the videos

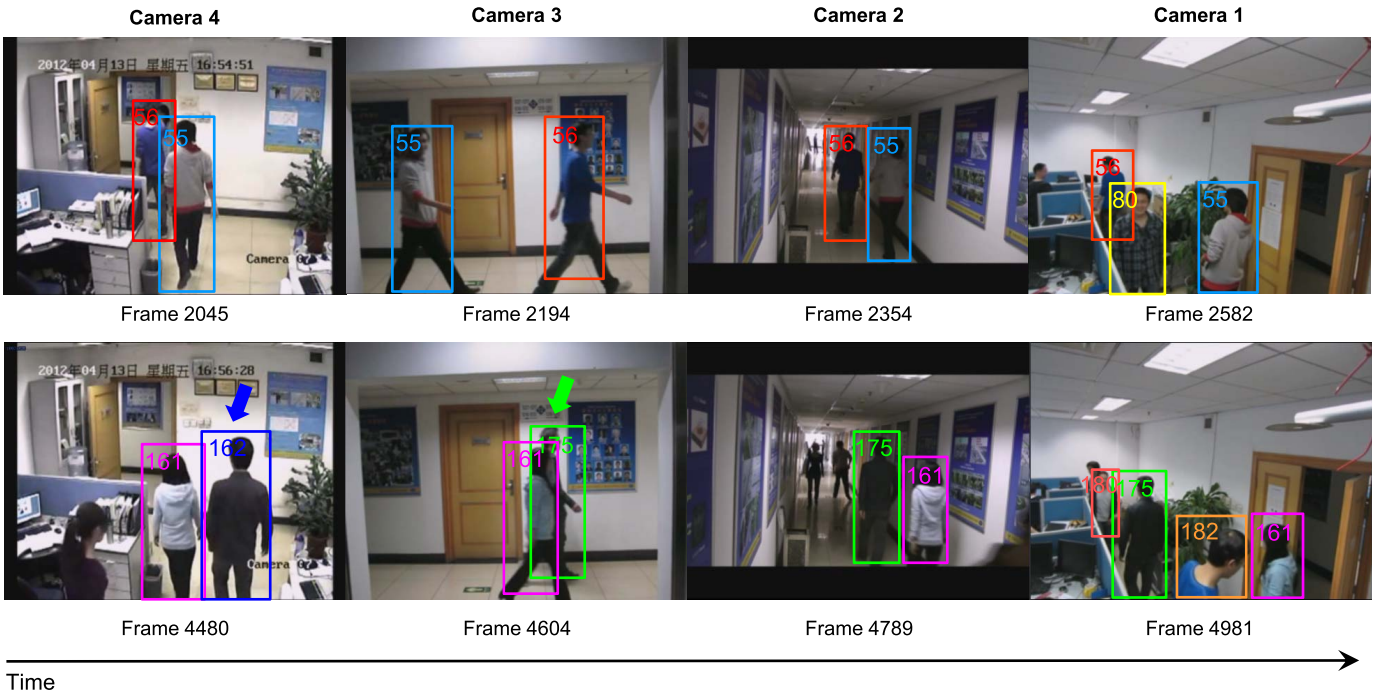


Fig. 10. Sample tracking results of our proposed method on Dataset3. In the first row, by taking advantage of the grouping information, targets 55 and 56 are successfully tracked in all cameras, even under significant within- and across-camera illumination changes. In the second row, target 162 in Camera 4 is not correctly linked to the same target (target 175) in Camera 3. This target is severely occluded by target 161 in Camera 3, even with group information we are unable to link them, as such association does not maintain appearance consistency. Bounding box with the same color indicates the same target. Best viewed in color.

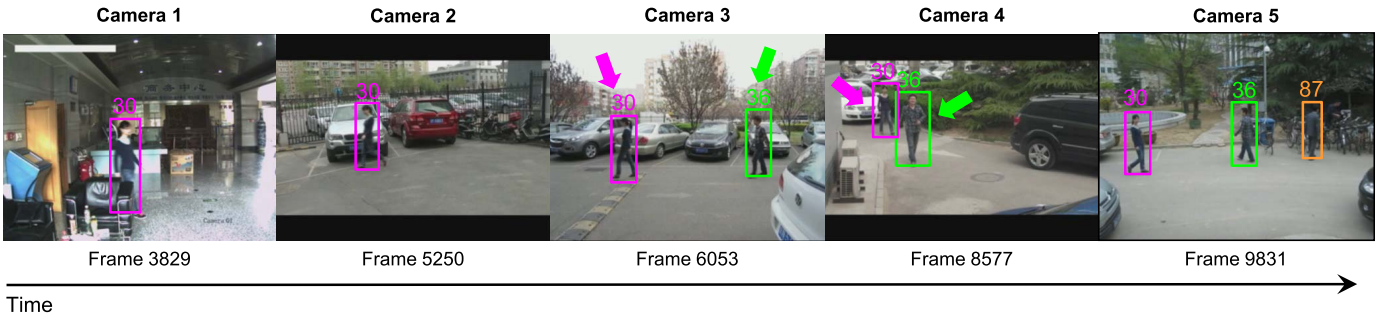


Fig. 11. Sample tracking results of our proposed method on Dataset4. Target 30 is correctly tracked in all cameras. Target 36 in Camera 3 is mistakenly linked to another target in Camera 4 (pointed by green arrow). This error occurs because the two targets with green arrow in Cameras 3 and 4 have group consistency (both of them form an elementary group with target 30) and appearance consistency (they are visually very similar). Bounding box with the same color indicates the same target. Best viewed in color.

in Dataset4 and achieves the highest MCTA on this subdata set. For Dataset1 and Dataset2, where there is less grouping information as compared with Dataset4, our proposed model has the second highest MCTA compared with the state of the art.

Due to the narrow view point for cameras in Dataset3 (see Fig. 10), each target enters and exits the scene in a very short time. It is difficult to detect elementary groups, as two targets can form an elementary group only if they co-exist for at least 2 s in our experiments. In Dataset3, the median length of all tracks is 3.9 s, while in other subdata sets the median length is at least 5.5 s. Therefore, among all the four subdata sets, the proposed method has the lowest MCTA on this data set. Although current setting of parameters for elementary group learning may not be the best choice for all subdata sets, it has been shown to be reasonable, as the tracking performance of the proposed method has better or comparable performance on all subdata sets compared with the state of the art. In order to learn a set of better parameters, more prior knowledge about the scene and the traffic would be necessary. This could be an interesting future direction for this paper.

Furthermore, to validate the robustness of the proposed method against imperfect input, estimated single camera tracking results are used for track association across cameras. Some previous work handles tracking using a species-based particle swarm optimization algorithm [48]. In this paper, the multitarget tracking model in [8] is employed to generate single camera tracks for our method. In this single camera tracking, occlusion is handled through tracklet association. Given a video sequence, a human detector is, first, applied to each frame to produce detection responses. For video frames where occlusion among targets exists, the occluded

TABLE IV
COMPARISON OF CROSS-CAMERA TRACKING RESULTS ON THE NLPR_MCT DATA SET WITH ESTIMATED SINGLE CAMERA TRACKING RESULTS AS INPUT

Method	Dataset1			Dataset2			Dataset3			Dataset4		
	mme^s	mme^c	MCTA	mme^s	mme^c	MCTA	mme^s	mme^c	MCTA	mme^s	mme^c	MCTA
This paper	101	61	0.8162	177	92	0.7730	149	133	0.1240	103	137	0.4637
USC-Vision [24]	63	35	0.8831	61	59	0.8397	93	111	0.2427	70	141	0.4357
Hfutdspmct [45]	77	84	0.7477	109	140	0.6561	105	121	0.2028	97	188	0.2650
CRIPAC-MCT [45]	135	103	0.6903	230	153	0.6234	147	139	0.0848	140	209	0.1830

targets have missed detections. Then, detection responses in consecutive frames that have a high probability to contain the same person are associated with form tracklets (trajectory fragments). Association of detections is based on visual similarity and overlap between the detection bounding boxes. Afterward, a global optimization method is employed to associate tracklets according to multiple cues, such as appearance, time, motion, and social grouping behavior. Finally, the missed detections are inserted by interpolation between linked tracklets.

Experimental results of our proposed model and the state-of-the-art approaches with estimated single camera tracking results as input are shown in Table IV. Note that different single camera tracking methods are used for the cross-camera tracking models compared in Table IV. Comparing the results from Tables III and IV, it is obvious that the quality of input tracks has a great impact on the performance of multicamera tracking. In general, the tracking performance of all tested methods degrade when imperfect single camera tracks are used as input. According to the results in Table IV, the proposed method still achieves better or comparable performance on all subdata sets when compared with the state of the art, which proves the robustness of our model against noise input.

V. CONCLUSION

In this paper, we presented a novel CRF model-based framework for multitarget tracking across a network of nonoverlapping cameras. The proposed model is able to systematically integrate social grouping behavior as the high-level contextual information for reducing ambiguities in track association. Experiments on four challenging real-world data sequences validated the effectiveness of our model. When there is rich grouping information in the scene, the tracking performance is significantly improved with the learned high-level contextual information. Possible future work would be to learn more discriminative representations for the targets.

APPENDIX

It has been shown in [44] that a CRF model is exactly solvable by graph cut in polynomial time, if the CRF model only contains potentials of up to two variables and the energy function satisfies submodularity. Submodularity is maintained if for each pairwise term B^{ij}, the following condition holds:

$$B^{ij}(1,0) + B^{ij}(0,1) \geq B^{ij}(1,1) + B^{ij}(0,0). \qquad (16)$$

Since there might exist conflicting edges in our proposed CRF model, we do not allow assignment $(1, 1)$ for such edges. In other words, for a conflicting edge, its corresponding edge cost of $B(1, 1) = \infty$. Therefore, (16) cannot hold for all pairwise terms in our energy function, and graph cut is not applicable.

REFERENCES

[1] N. S. Sthakoor, L. An, B. Bhanu, S. Sunderrajan, and B. S. Manjunath, "People tracking in camera networks: Three open questions," *Computer*, vol. 48, no. 3, pp. 78–86, Mar. 2015.

[2] B. Bhanu, C. V. Ravishankar, A. K. Roy-Chowdhury, H. Aghajan, and D. Terzopoulos, Eds., *Distributed Video Sensor Networks*. London, U.K.: Springer, 2011.

[3] I. Saleemi, K. Shafique, and M. Shah, "Probabilistic modeling of scene dynamics for applications in visual surveillance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 8, pp. 1472–1485, Aug. 2009.

[4] W. Hu, M. Hu, X. Zhou, T. Tan, J. Luo, and S. Maybank, "Principal axis-based correspondence between multiple cameras for people tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 663–671, Apr. 2006.

[5] X. Wang, "Intelligent multi-camera video surveillance: A review," *Pattern Recognit. Lett.*, vol. 34, no. 1, pp. 3–19, 2013.

[6] N. T. Siebel and S. J. Maybank, "The ADVISOR visual surveillance system," in *Proc. Eur. Conf. Comput. Vis. Workshop (ECCVW)*, 2004, pp. 103–111.

[7] *NLPR MCT Dataset*. [Online]. Available: http://mct.idealtest.org/Datasets.html

[8] X. Chen, Z. Qin, L. An, and B. Bhanu, "An online learned elementary grouping model for multi-target tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 1242–1249.

[9] M. Moussaïd, N. Perozo, S. Garnier, D. Helbing, and G. Theraulaz, "The walking behaviour of pedestrian social groups and its impact on crowd dynamics," *PLoS ONE*, vol. 5, no. 4, p. e10047, Apr. 2010.

[10] W. Ge, R. T. Collins, and R. B. Ruback, "Vision-based analysis of small groups in pedestrian crowds," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 1003–1016, May 2012.

[11] D. Helbing and P. Molnár, "Social force model for pedestrian dynamics," *Phys. Rev. E*, vol. 51, pp. 4282–4286, May 1995.

[12] T.-H. Chang and S. Gong, "Tracking multiple people with a multi-camera system," in *Proc. IEEE Workshop Multi-Object Tracking*, 2001, pp. 19–26.

[13] J. Black, T. Ellis, and P. Rosin, "Multi view image surveillance and tracking," in *Proc. Workshop Motion Video Comput.*, Dec. 2002, pp. 169–174.

[14] A. Gilbert and R. Bowden, "Tracking objects across cameras by incrementally learning inter-camera colour calibration and patterns of activity," in *Proc. 9th Eur. Conf. Comput. Vis. (ECCV)*, 2006, pp. 125–136.

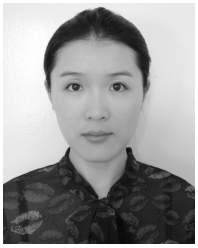
[15] O. Javed, K. Shafique, Z. Rasheed, and M. Shah, "Modeling inter-camera space–time and appearance relationships for tracking across non-overlapping views," *Comput. Vis. Image Understand.*, vol. 109, pp. 146–162, Feb. 2008.

[16] B. Prosser, S. Gong, and T. Xiang, "Multi-camera matching using bi-directional cumulative brightness transfer functions," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, Sep. 2008, p. 164.1.

[17] C.-T. Chu, J.-N. Hwang, K.-M. Lan, and S.-Z. Wang, "Tracking across multiple cameras with overlapping views based on brightness and tangent transfer functions," in *Proc. 5th ACM/IEEE Int. Conf. Distrib. Smart Cameras (ICDSC)*, Aug. 2011, pp. 1–6.

[18] T. D'Orazio, P. L. Mazzeo, and P. Spagnolo, "Color brightness transfer function evaluation for non overlapping multi camera tracking," in *Proc. 3rd ACM/IEEE Int. Conf. Distrib. Smart Cameras (ICDSC)*, Aug./Sep. 2009, pp. 1–6.

[19] S. Srivastava, K. K. Ng, and E. J. Delp, "Color correction for object tracking across multiple cameras," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2011, pp. 1821–1824.
[20] S. D. Hristov, "Multi-target tracking in unevenly illuminated scenes," Ph.D. dissertation, Univ. Trento, Trento, Italy, 2015.
[21] C.-H. Kuo, C. Huang, and R. Nevatia, "Inter-camera association of multi-target tracks by on-line learned appearance affinity models," in *Proc. 11th Eur. Conf. Comput. Vision I*, 2010, pp. 383–396.
[22] Y. Gao, R. Ji, L. Zhang, and A. Hauptmann, "Symbiotic tracker ensemble toward a unified tracking framework," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 7, pp. 1122–1131, Jul. 2014.
[23] S. Zhang, Y. Zhu, and A. Roy-Chowdhury, "Tracking multiple interacting targets in a camera network," *Comput. Vis. Image Understand.*, vol. 134, pp. 64–73, May 2015.
[24] Y. Cai and G. Medioni, "Exploring context information for inter-camera multiple target tracking," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2014, pp. 761–768.
[25] X. Chen, K. Huang, and T. Tan, "Object tracking across non-overlapping views by learning inter-camera transfer models," *Pattern Recognit.*, vol. 47, no. 3, pp. 1126–1137, 2014.
[26] I. Kviatkovsky, A. Adam, and E. Rivlin, "Color invariants for person reidentification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1622–1634, Jul. 2013.
[27] Y. Yang, J. Yang, J. Yan, S. Liao, D. Yi, and S. Z. Li, "Salient color names for person re-identification," in *Proc. 13th Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 536–551.
[28] C. Liu, S. Gong, and C. C. Loy, "On-the-fly feature importance mining for person re-identification," *Pattern Recognit.*, vol. 47, no. 4, pp. 1602–1615, Apr. 2014.
[29] R. Zhao, W. Ouyang, and X. Wang, "Learning mid-level filters for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 144–151.
[30] W.-S. Zheng, S. Gong, and T. Xiang, "Reidentification by relative distance comparison," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 3, pp. 653–668, Mar. 2013.
[31] F. Xiong, M. Gou, O. Camps, and M. Sznaier, "Person re-identification using kernel-based metric learning methods," in *Proc. 13th Eur. Conf. Comput. Vis.*, 2014, pp. 1–16.
[32] L. An, M. Kafai, S. Yang, and B. Bhanu, "Person reidentification with reference descriptor," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 4, pp. 776–787, Apr. 2016.
[33] L. An, S. Yang, and B. Bhanu, "Person re-identification by robust canonical correlation analysis," *IEEE Signal Process. Lett.*, vol. 22, no. 8, pp. 1103–1107, Aug. 2015.
[34] D. Tao, L. Jin, Y. Wang, and X. Li, "Person reidentification by minimum classification error-based KISS metric learning," *IEEE Trans. Cybern.*, vol. 45, no. 2, pp. 242–252, Feb. 2015.
[35] X. Chen, L. An, and B. Bhanu, "Multitarget tracking in nonoverlapping cameras using a reference set," *IEEE Sensors J.*, vol. 15, no. 5, pp. 2692–2704, May 2015.
[36] B. Yang and R. Nevatia, "An online learned CRF model for multi-target tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 2034–2041.
[37] A. Heili, C. Chen, and J. Odobez, "Detection-based multi-human tracking using a CRF model," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCV Workshops)*, Nov. 2011, pp. 1673–1680.
[38] E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Comput. Graph. Appl.*, vol. 21, no. 5, pp. 34–41, Sep./Oct. 2001.
[39] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY, USA: Wiley, 1991.
[40] X.-Y. Wang, J.-F. Wu, and H.-Y. Yang, "Robust image retrieval based on color histogram of local feature regions," *Multimedia Tools Appl.*, vol. 49, no. 2, pp. 323–345, 2010.
[41] X. Liu and D. Wang, "Texture classification using spectral histograms," *IEEE Trans. Image Process.*, vol. 12, no. 6, pp. 661–670, Jun. 2003.
[42] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[43] J. van de Weijer, C. Schmid, J. Verbeek, and D. Larlus, "Learning color names for real-world applications," *IEEE Trans. Image Process.*, vol. 18, no. 7, pp. 1512–1523, Jul. 2009.
[44] V. Kolmogorov and R. Zabin, "What energy functions can be minimized via graph cuts?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 147–159, Feb. 2004.
[45] *Multi-Camera Object Tracking (MCT) Challenge*. [Online]. Available: http://mct.idealtest.org/index.html
[46] A. Milan, K. Schindler, and S. Roth, "Challenges of ground truth evaluation of multi-target tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2013, pp. 735–742.
[47] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," *J. Image Video Process.*, vol. 2008, Feb. 2008, Art. no. 1.
[48] X. Zhang, W. Hu, W. Qu, and S. Maybank, "Multiple object tracking via species-based particle swarm optimization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 11, pp. 1590–1602, Nov. 2010.



Xiaojing Chen received the B.S. degree in information management and information systems from Beijing Language and Culture University, Beijing, China, in 2007; the M.S. (Hons.) degree in computer science from Leiden University, Leiden, The Netherlands, in 2009; and the Ph.D. degree in computer science from University of California at Riverside, Riverside, CA, USA, in 2015.

Her recent research has been concerned with multitarget tracking in surveillance cameras. Her current research interests include computer vision, pattern recognition, and machine learning.

Bir Bhanu (F'95) received the B.S. (Hons.) degree in electronics engineering from IIT Varanasi, Varanasi, India; the M.E. (Hons.) degree in electronics engineering from Birla Institute of Technology and Science, Pilani, India; the S.M. and E.E. degrees in electrical engineering and computer science from Massachusetts Institute of Technology, Cambridge, MA, USA; the Ph.D. degree in electrical engineering from University of Southern California, Los Angeles, CA, USA; and the M.B.A. degree from University of California at Irvine, Irvine, CA, USA.

He was a Senior Honeywell Fellow with Honeywell Inc., Minneapolis, MN, USA. He was the Founding Professor of Electrical Engineering with the University of California at Riverside (UCR), Riverside, CA, USA, where he served as the First Chair from 1991 to 1994. He has been the Cooperative Professor of CSE since 1991, of Bioengineering since 2006, and of ME since 2008, and the Director of the Visualization and Intelligent Systems Laboratory since 1991. He has been a Faculty Member with the Computer Science Department, University of Utah, Salt Lake City, UT, USA, Ford Aerospace and Communications Corporation, Newport Beach, CA, USA, the French Institute for Research in Computer Science and Automation, Rocquencourt, France, and the IBM San Jose Research Laboratory, San Jose, CA, USA. He has been the PI of various programs for the NSF, DARPA, NASA, the Air Force Office of Scientific Research, the Office of Naval Research, the Army Research Office, and other agencies and industries in the areas of video networks, video understanding, video bioinformatics, learning and vision, image understanding, pattern recognition, target recognition, biometrics, autonomous navigation, image databases, and machine-vision applications. He is currently the Distinguished Professor of Electrical Engineering and serves as the Founding Director of the Interdisciplinary Center for Research in Intelligent Systems with UCR. In addition, he serves as the Director of the NSF Interdisciplinary Graduate Education, Research and Training Program on Video Bioinformatics and the Interim Chair of the Department of Bioengineering at UCR. He has co-authored seven books and edited four books. He has authored 500 reviewed technical publications, including over 140 journal papers, 44 book chapters, and 15 patents.

Dr. Bhanu is a fellow of the American Association for the Advancement of Science, the International Association of Pattern Recognition, the American Institute for Medical and Biological Engineering, and the International Society for Optics and Photonics. He has been a recipient of many Best Conference Papers and Outstanding Journal Paper Awards and the Industrial and University Awards for Research Excellence, Outstanding Contributions, and Team Efforts and the Doctoral/Dissertation Advisor/Mentor Award. He has been on the Editorial Board of many journals and has edited, as the Lead Editor, the special issues of over 15 journals, including some of the top IEEE publications. He was the General Chair of the IEEE CVPR, AVSS, ICDSC, WACV, and DARPA IUW. Recently, he served on the IEEE Fellow Committee from 2010 to 2012.