Multi-camera Pedestrian Tracking using Group Structure

Zhixing Jin
Center for Research in Intelligent Systems
University of California, Riverside
900 University Ave, Riverside, CA 92507
jinz@cs.ucr.edu

Bir Bhanu
Center for Research in Intelligent Systems
University of California, Riverside
900 University Ave, Riverside, CA 92507
bhanu@cris.ucr.edu

ABSTRACT

Pedestrian tracking has been a popular research topic and application in the field of computer vision. Recently group information has been receiving increasing attention for pedestrian tracking, especially in highly occluded scenarios that make traditional vision features unreliable. In this paper, we propose a novel multi-camera pedestrian tracking system which incorporates a pedestrian grouping strategy and an online cross-camera model. The new cross-camera model is able to take the advantage of the information from all camera views as well as the group structure in the inference stage, and can be updated based on the learning approach from structured SVM. The experimental results demonstrate the improvement in tracking performance when grouping stage is integrated.

Categories and Subject Descriptors

I.4.8 [**Image Processing and Computer Vision**]: Scene Analysis—*Tracking*

General Terms

Algorithms

Keywords

Pedestrian tracking, Multi-camera, Group structure

1. INTRODUCTION

During the past decades, pedestrian tracking has always kept its popularity among various topics in the field of computer vision. It has made crucial contributions to many important application areas such as security surveillance and resource management. However, due to many existing challenges, the problem is still far from getting a perfect solution, although there have been a large amount of approaches invented and applied [1, 3, 5, 6, 8, 9, 13, 16].

One of the biggest challenges is the occlusions of pedestrians in the scene. For each pedestrian, the occlusion can be introduced by static objects in surrounding environment (e.g., buildings, trees), as well as other pedestrians in the same scene, especially when the density of the pedestrians in the scene gets higher. The severe occlusions can easily make the appearance and even shape models, which are the most important components that traditional tracking approaches are based on, no longer reliable. To overcome the occlusion problem, researchers proposed a variety of different methods. For example, training models for heads and/or different parts of human body [3, 17], using top view and depth information to avoid possible occlusion between pedestrians [15], constructing trajectories in a certain sliding window which provides both past and future information [3, 6, 16], or using multiple cameras with overlapping field-of-views (FOVs) for additional information [9, 14]. In this paper, since we are focused on pedestrian tracking problem with medium density, which may lead to severe occlusions in a single camera, a system consisting of cameras with overlapping FOVs is applied to combine the information from multiple views.

Another challenge comes from the similar appearances of pedestrians. This refers to not only the similar shape of pedestrians shown in the captured images, but also the colors of the clothes in many cases especially when severe occlusions exist. Up to now, the usage of spatial and temporal information is the main strategy for solving this problem. This includes but is not limited to: applying distance threshold when connecting detections from consecutive frames to form tracklets [3, 16], grouping nearby pedestrians with similar velocity together and tracking with the help of their averaged trajectory [6, 16], and generating confidence masks when associate detections and trackers [5]. In this paper, however, a more decent structure preserving object tracking (SPOT) approach, which further takes into account the graph property of the tracking objects [18] is considered. Either the minimum spanning tree or the central position of the graph is maintained and it is used in determining the location of the whole graph. The approach is originally designed for general object tracking and the spatial relationship is computed across all objects, but in our system, we added a grouping stage based on [6] and the spatial relationship is only effective inside each group.

The framework for the proposed approach is illustrated in Figure 1. For each pedestrian, an SVM classifier is trained for each camera view during the initialization stage. At each time step, the pedestrians are grouped into different groups based on their locations and velocities [6]. Then a com-

ICDSC '14, November 04 - 07 2014, Venezia Mestre, Italy
Copyright 2014 ACM 978-1-4503-2925-5/14/11...$15.00.
http://dx.doi.org/10.1145/2659021.2659023

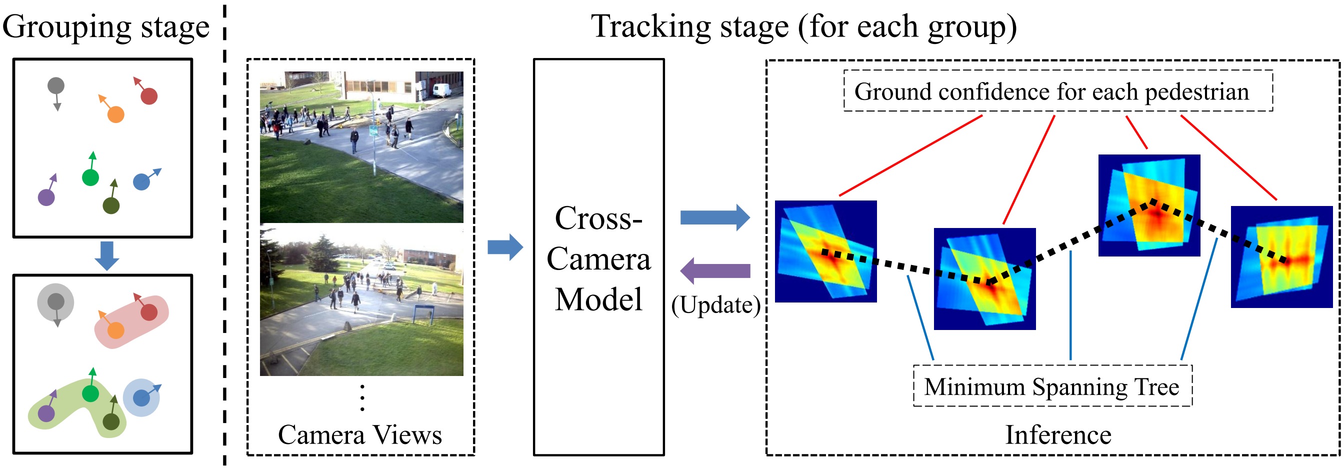

Figure 1: The system framework. At each time step, pedestrians are firstly grouped together (grouping stage), then the tracking is performed on each group (tracking stage).

bined confidence map for the whole group is computed on the ground plane based on the classifiers as well as the group structure. The tracking is conducted on this confidence map and the ground location for each pedestrian in the group is estimated. Finally, the estimated ground location for each pedestrian is used to update the cross-camera pedestrian model if necessary. This framework is similar to the SPOT tracker [18], with extension to multiple cameras. Furthermore, it takes the advantages of group structure information in multi-camera tracking and is expected to outperform the tracking approach without taking into account the group information.

2. RELATED WORK AND CONTRIBUTIONS

2.1 Related Work

The *state-of-the-art* tracking approaches are mostly based on sophisticated classification and/or detection methods, and they are known as tracking-by-classification or tracking-by-detection. In addition, these tracking approaches are designed in an online manner, which means that only limited initialization information is required. For example, both of the online Ada-Boosting [10] and Multiple Instance Learning [2] trackers use the online boosting classifier and can be initialized from the first frame. The general idea for this type of tracker is to train an online classifier during the initialization stage based on the limited information provided. Then this classifier is used to classify and locate the object from background. In addition, because the model for the object can be changed as time varies (e.g., illumination change, occlusions, etc), the classifier should have the ability to update its model based on the recent tracking result. The most widely used features for this type of classifiers include various appearance features as well as shape features [1, 2, 10, 11, 18]. They have been proven by a lot of successes in solving many general object tracking, especially single object tracking problems.

However, as pedestrian tracking usually has far more than one object (pedestrian) in the same scenario, occlusions may occur more frequently and similar pedestrian characteristics may be more and more shared among different objects. This makes the appearance and shape features not reliable enough to classify pedestrians from the background, or to distinguish one pedestrian from another. In this case, the spatial and temporal relationships between objects (pedestrians) begin to show their importance. For pedestrian tracking problems, one of the simplest but most representative relationships is the group information. Different from general objects, pedestrians have social property and, therefore, a crowd of pedestrians will be naturally formed into groups. It has also been shown that the integration of group information would provide positive feedback for the performance of pedestrian tracking. In the work from [6, 16], the data association between tracklets is significantly improved after the introduction of a grouping stage. Even for the general object tracking, group structure can also be very helpful, especially when these objects have similar appearances [18].

Another way to improve multi-person tracking performance is to use camera networks. Camera networks basically have two settings: overlapping [8, 9, 14] and non-overlapping [6, 13], or sometimes the mixture of them. But only the overlapping setting can provide additional information for the pedestrians in the same scene, and thus, improve the tracking performance. When pedestrians occur under a camera system with multiple cameras with overlapping field-of-views (FOVs), the same scenario may appear completely differently as the perspectives of cameras differ from each other. As a result, the occlusions observed by different cameras may also be totally distinct, and for each pedestrian, the probability that it can be confidently tracked in at least one camera will become significantly higher. Of course, the data association between cameras is another complicated problem. Since we are focused on how group information influence multi-camera pedestrian tracking performance, we simply use fixed (manual) data association in our system to avoid errors induced by this problem.

2.2 Contributions of This Paper

The key contributions of this paper are: 1) The tracking system incorporates a grouping stage, and for pedestrians in the same group, the tracking is performed with the group structure. 2) The original SPOT tracker is modified and extended to be suitable for an environment with multiple cameras with overlapping FOVs. For each pedestrian, the information from all camera views is fused together in the ground plane for tracking, and the tracked location is then projected back to all views to update the tracker, if necessary.

3. TECHNICAL APPROACH

The system mainly consist of two stages: (1) a grouping stage which computes the groups for all the pedestrians based on their location and velocity information, and (2) a tracking stage which is able to track pedestrians in the same group. In this section, we will provide details for both of the two components.

3.1 Grouping

In the grouping stage, the pedestrians will be grouped based on their current status. For each pedestrian P_i, we maintain its status as (x^g, y^g, u^g, v^g) on the ground plane, where (x^g, y^g) is the ground location coordinate and (u^g, v^g) is the velocity. Similar to [6], a pair-wise grouping score is computed between every two pedestrians according to the relationship between their locations and velocities

$$S^g_{ij} = D^g_{ij} \cdot V^g_{ij} \tag{1}$$

where D^g and V^g are the scores based on the distance and the velocity between them, respectively, which are calculated using the equations

$$D^g_{ij} = 1 - \frac{2}{\pi} \arctan(dist(P_i, P_j)) \tag{2}$$

$$V^g_{ij} = 1 + \frac{\vec{v_i} \cdot \vec{v_j}}{\|\vec{v_i}\| \cdot \|\vec{v_j}\|} \tag{3}$$

Here, $dist(P_i, P_j)$ is a relative distance between the two pedestrians and $\vec{v_i}$, $\vec{v_j}$ are their velocities. Unlike the distance used in [6], we use a simpler Euclidean-based distance since the pedestrian status in our system include their locations on the ground plane

$$dist(P_i, P_j) = \max\left(0, \frac{\|\vec{p_i} - \vec{p_j}\|}{r_i + r_j} - 1\right) \tag{4}$$

$\vec{p_i}$ and $\vec{p_j}$ are the ground location vectors for the two pedestrians, and r_i, r_j are the radius for them. In our system, we assume all pedestrians have the same radius (size), that is, $r_i = r_j = r$. According to this definition, the minimal value of the actual Euclidean distance between any two pedestrians is the sum of their radii ($2r$ in our system), thus $0 < D^g \leq 1$.

Based on these pair-wise grouping scores, we firstly label two pedestrians P_i and P_j to be in the same group when their grouping score $S^g_{ij} \geq T_g$ where T_g is the threshold for the grouping strategy and it is set to a fixed value during the whole experiment. Secondly, we extend these pair-wise grouping relationship to all the pedestrians by transitivity. That is, if P_i and P_j are labeled in the same group and P_i and P_k are labeled in the same group, then we label P_j and P_k to be in the same group as well. This step iterates until a stable result is achieved (convergence). Finally, the set of all pedestrians $\mathbb{P} = \{P_i\}$ can be divided into a set of groups (partitions) $\mathbb{G} = \{G_i\}$, so that each group contains a set of pedestrian indexes $G_i = \{i_1, i_2, \ldots, i_n\} (1 \leq i_k \leq |\mathbb{P}|)$ and none of the two groups are overlapping $G_i \cap G_j = \emptyset (\forall i, j)$. The tracking approach in the following section is then conducted based on the group partitions.

3.2 Tracking

The tracking stage is conducted on each group based on the grouping result $\mathbb{G}$. This stage has three general steps: (1) compute a confidence map on the ground plane for each pedestrian; (2) tracking pedestrians in each group according to their ground confidence maps; (3) for each pedestrian, update the relevant classifiers as well as the frame locations for all camera views. To accomplish this tracking task, we modified and extended the idea from [18] to make it suitable for multi-camera tracking.

3.2.1 Cross-Camera Model

The pedestrian model in the original work [18] requires modification to maintain the information from all camera views. For each pedestrian P_i, we define a bounding box $B^v_i = (x^v_i, y^v_i, w^v_i, h^v_i)$ for each camera view $v (v \in \mathbb{V})$, where (x^v_i, y^v_i) is the frame location and (w^v_i, h^v_i) is the size information (width and height).

In a pedestrian tracking system, the camera views are usually perspective, which makes the change in size of the bounding box relevant to the frame location changing. Therefore, we can use only two variables (x^v_i, y^v_i) to get both of the most recent position and size information of the bounding box. In addition, to simplify this processing and unify the features across all pedestrians, we used a scaling technique for the bounding box sizes. We set a target size for all pedestrian bounding boxes as (w, h) and compute the scale as $l^v_i = h^v_i / h$. Then, given the initial scale of a bounding box $\tilde{l^v_i}$, the status for a bounding box B^v_i in our system can be re-defined as $B^v_i \equiv \tilde{B^v_i}$, where $\tilde{B^v_i} = (x^v_i, y^v_i, \tilde{l^v_i})$.

A configuration for all the pedestrians in a group $G \in \mathbb{G}$ is then defined as the set of all their bounding boxes for all camera views. Using the notation of unified bounding boxes and scales, $\mathbb{C} = \{\tilde{B^v_i}\} (\forall v \in \mathbb{V}, \forall i \in G)$.

The feature used in our system is the histogram of gradient (HOG), first proposed by [7], which is the same feature used in the original work. For each bounding box $\tilde{B^v_i}$ and a frame I_v, we firstly compute the current scale l^v_i of the bounding box according to its position, and then re-size the frame by this scale. This ensures that the bounding box is re-sized to the target size (w, h). The HOG feature is then extracted on the re-sized frame. We use $\phi(I_v; \tilde{B^v_i})$ to denote the feature extraction. The output for the function $\phi(\cdot)$ is a concatenated feature vector.

Then for each pedestrian P_i and camera view v, the confidence can be calculated as

$$c^f_v(C^v_i; I_v, \mathbf{w}^v_i) = \mathbf{w}^{vT}_i \cdot \phi(I_v; \tilde{B^v_i}) \tag{5}$$

where $\mathbf{w}^v_i$ is the weight vector on the HOG features extracted from unified bounding boxes, and C^v_i is the configuration for this particular pedestrian and camera view, $C^v_i = \{\tilde{B^v_i}\}$. Then the ground confidence can be computed using the following equation

$$c^g(C_i; \mathbb{I}, \theta_i) = \frac{1}{\|\mathbb{V}\|} \sum_{v \in \mathbb{V}} \mathcal{H}_v \left(c^f_v(C^v_i; I_v, \mathbf{w}^v_i) \right) \tag{6}$$

where $\mathbb{I}$ is defined as the set of all frames across all views $\mathbb{I} = \{I_v\} (\forall v \in \mathbb{V})$, similarly, $C_i = \{C^v_i\} (\forall v \in \mathbb{V})$, $\theta_i = \{\mathbf{w}^v_i\} (\forall v \in \mathbb{V})$. $\mathcal{H}_v(\cdot)$ is a projection from the frame coordinates of camera view v to the coordinates on the ground plane.

Since the minimum spanning tree (MST) model is reported to have better performance in [18], we use only this model in our system, and an edge e_{ij} in the tree denotes that the two pedestrians P_i and P_j are connected and $e_{ij} = \vec{p_i} - \vec{p_j}$.

Therefore, for the complete configuration C, its corresponding score can be calculated as

$$S^c(\mathbb{C}; \mathbb{I}, \Theta) = \sum_{i \in G} c^g(C_i; \mathbb{I}, \theta_i) - \lambda \sum_{\mathcal{E}(i,j)=1} \|(\vec{p_i} - \vec{p_j}) - e'_{ij}\|^2 \tag{7}$$

where e'_{ij} is the edge in the minimum spanning tree and is computed according to the previous location information of all the pedestrians. $\mathcal{E}(i, j)$ is an indicator function which denotes whether there is an edge in the tree connecting pedestrians P_i and P_j. Θ is the set of all parameters, $\Theta = \{\theta_i\} \cup \{e_{ij}\} (\forall i \in G, \mathcal{E}(i, j) = 1)$. In our system, the MST is computed after the grouping stage, and e_{ij} is updated if necessary, but before the inference.

3.2.2 *Ground Inference*

The purpose of ground inference is to find the optimal configuration C^* for all the pedestrians in a group, which maximizes the configuration score. As stated in the original work, for a tree-structured graph, this optimization can be performed in linear time using a combination of dynamic programming and min-convolution [18].

In our system, the inference processing is the same as in the original work after we project the confidences from each view onto the ground plane and sum them up. The message-passing equations have the form as (starting from the root node)

$$R_{ij}(\vec{p_i}) = c^g(C_i; \mathbb{I}, \theta_i) + \sum_{\forall k \neq j : \mathcal{E}(k,i)=1} \mu_{k \to i}(\vec{p_i}) \tag{8}$$

$$\mu_{i \to j}(\vec{p_i}) = \max_{\vec{p_i}'} \left(R_{ij}(\vec{p_i}') - \lambda \|(\vec{p_i} - \vec{p_i}') - e_{ij}\|^2 \right) \tag{9}$$

These two equations are exactly the same as in [18], except for the confidence calculation. This message-passing starts from the root node of the MST, and the optimal configuration C^* can be obtained after a full forward-backward pass along the tree. Therefore, the same inference algorithm can be applied in our system after the ground confidence for each pedestrian is calculated.

The pedestrian location on the ground plane, $\vec{p_i}$, is computed according to its corresponding configurations, by using the principal-axis based correspondence [12]. In addition, realizing that a pedestrian only moves within a relatively small distance at each time step, we extract HOG features only on a small region around its bounding box for computational efficiency. Therefore, the global optimization is limited to a small region around its ground location $\vec{p_i}$ as well. The detailed information will be provided in Section 4.

3.2.3 *Cross-camera Learning*

The model updating, or model learning process is also different from the original work since it is a cross-camera process.

When a set of observations $\mathbb{I}$ are obtained, the optimal configuration $\mathbb{C}^*$ is determined by maximizing Equation (7). Note, according to the definition, the optimal configuration includes not only the position of the bounding boxes on all camera views, but also their optimal scales. This optimal configuration is then considered as a true positive example. Similar to the original work [18], a margin function $\Delta(\mathbb{C}, \mathbb{C}^*)$ is defined for the structured SVM

$$\Delta(\mathbb{C}, \mathbb{C}^*) = \sum_{v \in \mathbb{V}} \sum_{i \in G} \left(1 - \frac{\tilde{B_i^v} \cap \tilde{B_i^v}^*}{\tilde{B_i^v} \cup \tilde{B_i^v}^*} \right) \tag{10}$$

The function is limited as $0 \leq \Delta(\mathbb{C}, \mathbb{C}^*) \leq |\mathbb{V}| \cdot |G|$, where 0 can be reached if and only if $\mathbb{C} = \mathbb{C}^*$. Then the loss function of the structured SVM is defined as

$$\begin{aligned} &\mathcal{L}(\Theta; \mathbb{I}, \mathbb{C}^*) \\ &\quad = \max_{\mathbb{C}} \left(S^c(\mathbb{C}; \mathbb{I}, \Theta) - S^c(\mathbb{C}^*; \mathbb{I}, \Theta) + \Delta(\mathbb{C}, \mathbb{C}^*) \right) \end{aligned} \tag{11}$$

Although this loss function has a more complex approach to calculate S^c, it only contains a set of affine functions, without any quadratic terms. Therefore, the loss function in Equation (11) is still a convex function w.r.t the parameter set Θ, which is the same as in the original work. As stated in [18], the gradient of this loss function does not work very well since it may tend to provide uninformative directions. Therefore, the same modification as in the original work, which is only based on the confidence scores, is used to define the new search direction $\mathbf{p}$. This direction is then used to update the parameter set.

Different from the original work, our tracking system has multiple views. Thus, the parameters from all views are updated simultaneously. However, because the parameter set is a vector concatenating all weight vectors $\mathbf{w}$ and the MST edge information, the weight vectors for different views can be updated independently as long as the global information for $\mathcal{L}(\Theta; \mathbb{I}, \mathbb{C}^*)$ and $\mathbf{p}$ are provided. The initialization of the weight vector $\mathbf{w}_i^v$ for each view is conducted by training an SVM using the initial patch as positive sample and 50 randomly selected patches as negative samples.

In addition, since we have a grouping stage at each time step, the MST may be different from frame to frame. Therefore, if an edge e_{ij} from the previous MST is preserved to the current time step, then the updated value is kept. Otherwise, if an edge is new, then its value is initialized after the MST is obtained.

4. EXPERIMENTAL RESULTS

In this paper, we conducted experiments to investigate the performance of the proposed multi-camera pedestrian tracking system. This section describes details for our experimental settings, and the final output of the system.

4.1 Experimental Settings

The dataset used is PETS 2009 with medium density crowd (i.e., S2.L2). It originally contains 4 views, but View 3 has a huge tree in the center of the frame and View 4 suffers from frame rate instability, so only frames from View 1 and View 2 are used in our experiments. The ground-truth for each view is manually annotated for every 5 frames and interpolated in between. The ground-truth on the ground plane is then computed using principal-axis based correspondence [12]. The target size of each pedestrian patch is set to 64×128. As a result, the frames from each view are re-sized to 2560×1920 so that the smallest pedestrian in the frame appears in a comparable size to the target pedestrian patch size.

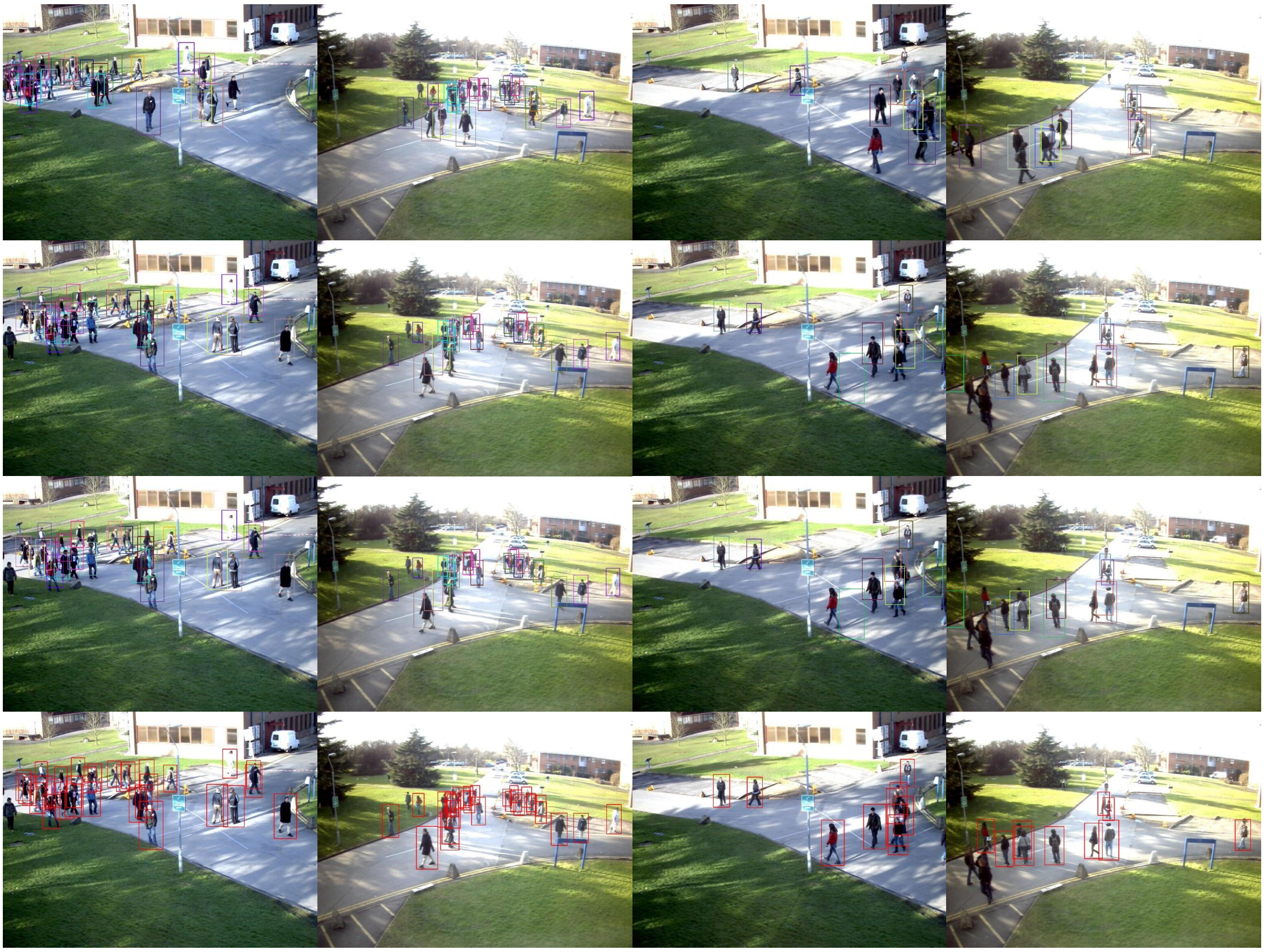

Figure 2: The results for the two segments. The left two columns are View 1 and View 2 for Segment 1, and the right two columns are for Segment 2. From top to bottom, the four rows illustrate the initial frames, the final frames without grouping, the final frames with grouping, and the ground-truth, respectively.

The ground plane is set to a grid with size of 700 × 700. The projection functions $\mathcal{H}_v(\cdot)$ are obtained by manually labeling four corresponding points across all views and the ground plane (via Google Maps). In addition, we check all the ground-truth of the ground locations for all the pedestrians to make sure that they are not exceeding the area of the ground grid. The radius of pedestrians r is set to 5, which is estimated from the ground plane size and the dataset frames.

For the parameters, we basically follow the settings from the original work [6, 18]. The grouping threshold T_g is set to 0.2 for all experiments. The λ in Equation (7) is set to 0.001. The confidence threshold T_p and the control parameter K in the model learning are set to 0.4 and 1, respectively.

For speed-up purposes, the inference for each pedestrian is only performed in a small region around its previous location. Therefore, on each frame, the search region is a small area around its previous frame location, with a size of 320 × 240. In addition, since the scale change inside this region is relatively small, we relax the constraint between the scale and the location change. That is, the confidence in the search region will be calculated over all possible scale levels. In the experiment, we use three scale levels: 0.95, 1, 1.05.

In this experiment, we select two segments of the dataset, with different densities. Segment 1 starts from frame #0, and Segment 2 starts from frame #300. Both of them have a length of 20 frames. There are about 30 pedestrians in the first segment and about 10 in the second one. The initial positive patches for each pedestrian come directly from the ground-truth. We tested two situations: with/without grouping. The results are reported in the next section.

4.2 Results

The metric used for evaluating the tracking performance is multi-object tracking precision (MOTP) and multi-object tracking accuracy (MOTA) [4].

For the frame results, since we have the bounding box information, the accuracy of the tracker is defined as the rectangle overlapping ratio between the tracked bounding box and the ground-truth, which is in a range of $[0, 1]$. When the ratio is above 0.5, we consider the tracked bounding box as accurate.

On the ground plane, the bounding boxes for pedestrians are no longer available. Thus, we use a Euclidean distance based accuracy calculation

$$d(tr, gt) = \max\left(0, 1 - \frac{\|p_{tr} - p_{gt}\|}{4r}\right) \quad (12)$$

The tracking is considered as accurate when $d(tr, gt) > 0.5$, that is, the distance between the tracked position and the ground-truth is less than the diameter of a pedestrian.

Table 1: The MOTP and MOTA for Segment 1 (Better results are shown in bold).

	No Grouping		Grouping	
	MOTP	MOTA	MOTP	MOTA
View 1	**71.10%**	62.73%	70.71%	**72.38%**
View 2	72.37%	70.38%	**72.52%**	**76.04%**
Ground plane	80.02%	52.08%	**80.44%**	**57.40%**

Table 2: The MOTP and MOTA for Segment 2 (Better results are shown in bold).

	No Grouping		Grouping	
	MOTP	MOTA	MOTP	MOTA
View 1	**71.75%**	55.25%	71.63%	**55.80%**
View 2	**73.12%**	**63.54%**	72.27%	**63.54%**
Ground plane	**83.10%**	51.38%	83.03%	**55.25%**

Table 1 and 2 show the MOTP and MOTA results for View 1, View 2, and the ground plane, for the two segments, respectively. The qualitative results are illustrated in Figure 2. The results reveal that the group information has a positive influence on the performance of the tracking system for medium density crowd with many occlusions (Segment 1). But in low density scenes, the integration of grouping does not lead to significant performance improvement, which is reasonable since less number of groups will be formed when people are moving sparsely.

5. CONCLUSIONS

In this paper, we proposed a new approach for multi-camera pedestrian tracking, which takes the advantage of group structures. It has a grouping stage and a tracking stage. For the grouping stage, the most recent approach which uses the location and velocity information for pedestrians is applied. For the tracking stage, a cross-camera model is set up for each pedestrian, which utilizes HOG features and maintains an SVM classifier for each camera view. The inference of this model is performed on the ground plane, based on the confidence map computed by the classifiers as well as the group structure. The model can be updated in an online manner as the tracking continues. The experimental results demonstrate that the group information has positive influence on the performance of the tracking system, especially when the density is relatively high.

6. ACKNOWLEDGMENTS

This work is supported in part by NSF grants 0905671 and 1330110.

7. REFERENCES

[1] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *CVPR '08*, June 2008.

[2] B. Babenko, M.-H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(8):1619–1632, Aug. 2011.

[3] B. Benfold and I. Reid. Stable multi-target tracking in real-time surveillance video. In *CVPR '11*, June 2011.

[4] K. Bernardin and R. Stiefelhagen. Evaluating multiple object tracking performance: the CLEAR MOT metrics. *J. Image Video Process.*, 2008:1:1–1:10, Jan. 2008.

[5] M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool. Online multiperson tracking-by-detection from a single, uncalibrated camera. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(9):1820–1833, Sept. 2011.

[6] X. Chen, Z. Qin, L. An, and B. Bhanu. An online learned elementary grouping model for multi-target tracking. In *CVPR '14*, 2014.

[7] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR '05*, 2005.

[8] W. Du and J. Piater. Multi-camera people tracking by collaborative particle filters and principal axis-based integration. In *ACCV '07*, 2007.

[9] R. Eshel and Y. Moses. Tracking in a dense crowd using multiple cameras. *Int. J. Comput. Vision*, 88(1):129–143, May 2010.

[10] H. Grabner and H. Bischof. On-line boosting and vision. In *CVPR '06*, 2006.

[11] S. Hare, A. Saffari, and P. H. S. Torr. Struck: Structured output tracking with kernels. In *ICCV '11*, Nov. 2011.

[12] W. Hu, M. Hu, X. Zhou, T. Tan, J. Lou, and S. Maybank. Principal axis-based correspondence between multiple cameras for people tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(4):663–671, Apr. 2006.

[13] 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 Und.*, 109(2):146 – 162, 2008.

[14] Z. Jin and B. Bhanu. Integrating crowd simulation for pedestrian tracking in a multi-camera system. In *ICDSC '12*, Oct. 2012.

[15] O. Ozturk, T. Yamasaki, and K. Aizawa. Tracking of humans and estimation of body/head orientation from top-view single camera for visual focus of attention analysis. In *ICCV Workshops '09*, Sept. 2009.

[16] Z. Qin. Improving multi-target tracking via social grouping. In *CVPR '12*, 2012.

[17] G. Shu, A. Dehghan, O. Oreifej, E. Hand, and M. Shah. Part-based multiple-person tracking with partial occlusion handling. In *CVPR '12*, June 2012.

[18] L. Zhang and L. van der Maaten. Preserving structure in model-free tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(4):756–769, Apr. 2014.