Harry Potter's Marauder's Map: Localizing and Tracking Multiple Persons-of-Interest by Nonnegative Discretization

Shoou-I Yu, Yi Yang, Alexander Hauptmann
Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA, 15213, USA
{iyu, yiyang, alex}@cs.cmu.edu

Abstract

A device just like Harry Potter's Marauder's Map, which pinpoints the location of each person-of-interest at all times, provides invaluable information for analysis of surveillance videos. To make this device real, a system would be required to perform robust person localization and tracking in real world surveillance scenarios, especially for complex indoor environments with many walls causing occlusion and long corridors with sparse surveillance camera coverage. We propose a tracking-by-detection approach with nonnegative discretization to tackle this problem. Given a set of person detection outputs, our framework takes advantage of all important cues such as color, person detection, face recognition and non-background information to perform tracking. Local learning approaches are used to uncover the manifold structure in the appearance space with spatio-temporal constraints. Nonnegative discretization is used to enforce the mutual exclusion constraint, which guarantees a person detection output to only belong to exactly one individual. Experiments show that our algorithm performs robust localization and tracking of persons-of-interest not only in outdoor scenes, but also in a complex indoor real-world nursing home environment.

1. Introduction

The *Marauder's Map*, which locates and tracks friends and enemies of Harry Potter in the magical world, is also invaluable in real world surveillance scenarios. If we are able to localize and track each person as shown in Figure 1, action recognition of people can be subsequently performed to analyze human behavior. To perform reliable localization and tracking, important cues such as color, person detection, face recognition and non-background detection should all be utilized. Also, the tracking algorithm has to deal with typical yet complex indoor scenes consisting of different rooms, many walls and corridors. Therefore, an ideal Marauder's Map algorithm should integrate different sources of information in a seamless way to perform reliable localization and tracking of persons-of-interest in complex indoor environments.

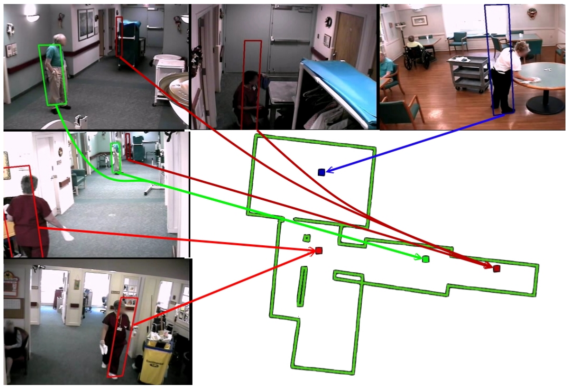


Figure 1. The Marauder's Map for a nursing home. The map of the nursing home is in the bottom right. Dots on the map show the location of a person-of-interest. The surrounding images are the views from each surveillance camera. Due to space limitations, only 5 out of 15 cameras are shown.

There are several cues available for trackers to utilize, such as color, person detection, face recognition and non-background detection. Each cue provides important information, but the cues are not always reliable. However, many existing tracking systems [3, 11, 1] only use a subset of cues to perform tracking. This will not be ideal when the cues used is not reliable in the current situation. Therefore, we propose an localization and tracking algorithm which utilizes color, person detection, face recognition and non-background detection cues to perform robust tracking. To the best of our knowledge, we are the first work to utilize face recognition for tracking.

Incorporating all the available cues seamlessly into a framework is not a trivial task. Our algorithm follows the tracking by detection paradigm [14, 1], which can handle re-initializations naturally and avoids excessive model drift. This paradigm is also less affected by occlusions caused by walls or sparse camera setups. Tracking by detection can be

1063-6919/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPR.2013.476

viewed as a classification problem. If we treat one person-of-interest as one class, the tracker needs to assign a class label to each person detection result. Sparse label information can be acquired from face recognition output and manually annotated start and end locations of a person. However, labels for most person detection results are unknown. These conditions motivate us to utilize semi-supervised learning techniques to perform multi-camera multi-object tracking. Given two person detection data points which are spatial-temporal neighbors, if they have similar appearances, it is very likely that the two points correspond to the same person. As this is a good fit to the manifold assumption, we propose to uncover the manifold structure of detected data points by leveraging local learning techniques. Inspired by [19], instead of directly computing the affinity matrix according to the features of data points, we adopt a statistical approach to exploit the manifold structure, which is more accurate and robust.

Simply satisfying the manifold assumption is not sufficient for reliable tracking. The mutual exclusion constraint, which constrains one person detection result to be associated with only one person, should also be embedded into the tracker. We perform *nonnegative discretization*, which partitions the detected data points into non-overlapping groups such that mutual exclusion and the manifold assumption are satisfied simultaneously. We formulate the problem as a nonnegative optimization problem. The optimization converges only to a local optima, and thus we resort to weak supervision for a more stable solution to initialize the algorithm.

In sum, the main contribution of this paper is as follows:

1. We propose a novel method which performs robust multiple person-of-interest localization and tracking by incorporating color, non-background, person detection and face recognition in a semi-supervised learning framework.
2. We perform experiments on a real-world complex indoor data set with long corridors and many walls causing occlusion. To the best of our knowledge, this is the first work that has applied multi-camera multi-object tracking in such a complex indoor environment.

The rest of the paper is organized as follows. After a brief review of multi-object tracking and semi-supervised tracking in Section 2, we will detail our algorithm in Section 3. Experiments are given in Section 4 and Section 5 concludes this paper.

2. Related Work

The Marauder's Map can be viewed as a multi-camera multi-object tracking problem. Multi-object tracking has been an active field of research for the past 10 years. Earlier work [14] models the multi-modal posterior with a set of samples and uses a color-based particle filter combined with an object detector to perform tracking. [13] uses Kalman filters to perform multi-person tracking with 16 cameras. Recent work discretizes the solution space of tracking and uses background and non-background information to locate potential objects to track for each frame. Graph-cuts [11] or k-shortest paths [3] are then used to perform optimization and find the trajectories of each object. Another line of work focuses on using the output of an object detector as input and using integer linear programming [10] or discrete-continuous energy minimization [1] to find trajectories. However, the aforementioned approaches only focus on using a subset of available information. The algorithms might not be robust if they just rely on a subset of cues, because some cues may be potentially noisy and unreliable. For example, trackers not using color information [3, 1] will have difficulty avoiding identity switches when multiple people come very close together and split up. Exploiting color information will help in disambiguating different people. Also, most of the aforementioned approaches use an unsupervised approach to perform tracking, which is convenient in that very little manual effort is required, but this will increase the amount of identity switches, which is not ideal for person-of-interest tracking. Finally, previous methods all perform experiments on wide and open indoor or outdoor scenes. It is unclear whether the performance of previous methods translates into complex indoor environments with sparse camera setups.

Recently, there have been papers focusing on leveraging semi-supervised learning to improve monocular tracking performance. [9] proposes an online boosting semi-supervised framework to find features that can effectively separate the tracking target from the background. The tracker relies on its own prediction scores to update its own model. [16] extends [9] by combining an offline object detector with an online object recognizer and an online semi-supervised tracker to alleviate the template drift problem. However, previous semi-supervised learning trackers are focused on *monocular* videos and do not take into account the interaction between multiple tracked objects. Our method is different and novel in that it uses semi-supervised learning to *jointly* learn the assignment of labels for all objects in a *multi-camera* environment.

3. Methodology

Following the tracking by detection paradigm [14], the input to our algorithm is a set of person detection results at each time instant. The person detection results from different camera views can be mapped to a common 3D coordinate system using camera calibration and ground plane parameters provided. Each person detection result is described by the color histogram of the person detection result. Our algorithm's main task is to predict a label for each person detection result. To perform the prediction task, our algorithm incorporates two main innovative components, which are manifold learning in appearance space with spatio-temporal constraints, and trajectory inference by nonnegative discretization. The following paragraphs will describe each step in detail.

3.1. Notations

Hereafter, we call a person detection result as a *data point*. Suppose there are n data points generated by the person detector. The color histogram for the i-th data point is denoted as $x_i \in \mathbb{R}^d$. Let p_i and t_i denote the 3D location and video frame of the i-th data point respectively. Let c be the number of individuals to be tracked. Our task is to assign each data point a class label. We denote $F = [f_1, f_2, \ldots, f_n]^T \in \mathbb{R}^{n\times c}$ as the scaled label indicator matrix of all the data points $1, 2, \ldots, n$ and $f_i \in \mathbb{R}^c$ is the label indicator vector for the i-th data point. Without loss of generality, we assume that the data points are reorganized such that the data points from the same class are put together. The j-th column of F is given by:

$$F_j = [\underbrace{0,\ldots,0}_{\sum_{i=1}^{j-1} n_i}, \underbrace{1,\ldots,1}_{n_j}, \underbrace{0,\ldots,0}_{\sum_{i=j+1}^{c} n_i}]^T/\sqrt{n_j}, \qquad (1)$$

where n_j is the number of data points in the j-th class. If the p-th element in F_j is $\frac{1}{\sqrt{n_j}}$, it indicates that the p-th data point corresponds to the j-th person. According to Equation 1, it can be verified that

$$F^T F = [F_1, \ldots, F_c]^T [F_1, \ldots, F_c] = I, \qquad (2)$$

where I is the identity matrix. In this paper, $Tr(\cdot)$ denotes the trace operator and $|\cdot|_F$ is the Frobenius norm of a matrix. Given an arbitrary number m, $\mathbf{1}_m \in \mathbb{R}^m$ is a column vector with all ones.

3.2. Manifold Learning in Appearance Space with Spatio-Temporal Constraints

The appearance of a person within a short period of time should not change much. Given two detected points which are spatio-temporal neighbors, if they have similar appearances, it is very likely that the two points correspond to the same person. As this is a good fit to the manifold assumption, we follow the method used in [19] to learn the manifold structure. Nearest neighbor selection is a crucial step in learning manifold structure. Therefore, in the following paragraphs, we will first detail the method we used for nearest neighbor selection and then describe how this information is utilized in manifold learning.

Given a data point, suitable neighbors are spatio-temporal neighboring data points with similar color histograms. For the i-th data point, let the set $\mathcal{S}_i$ contain data points which are not only less than T frames away from the point, but also reachable from location p_i with a reasonable velocity, *i.e.*,

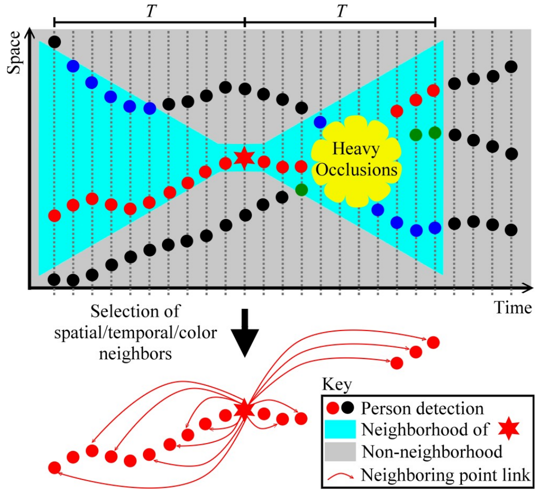


Figure 2. The neighbor selection method for data point ☆.

$$\mathcal{S}_i = \left\{ l \mid \frac{||p_i - p_l||_2}{|t_i - t_l|} \le V, |t_i - t_l| \le T, 1 \le l \le n \right\}, \qquad (3)$$

where V is the maximum possible velocity of a moving person. If the velocity required to move between two points is too large, then the two points cannot be of the same individual. Points in $\mathcal{S}_i$ are the spatio-temporal neighbors of the i-th data point, which are the points inside the bow-tie shaped region as shown in Figure 2. According to our observation, detections of the same individual in a local temporal neighborhood should have similar color histograms. Therefore, the nearest color histogram neighbors of the i-th data point in the spatio-temporal neighbor set $\mathcal{S}_i$ should correspond to data points from the same individual. We denote $N_i^k = [i, i_1, i_2, \ldots, i_k]$ as the vector containing the indices to the k nearest color histogram neighbors of the i-th data point in $\mathcal{S}_i$ and the index of i itself.

This method of finding neighbors is robust to occlusions due to the following reason. Occlusions may cause the tracking target to be partially or completely occluded. However, the tracking target usually reappears after a few frames. Instead of trying to explicitly model occlusions, we connect the observations of the tracking target before and after occlusion. As shown in Figure 2, despite heavy occlusions in a time segment, the algorithm is still able to link to the correct detections after the occlusion. The value of T affects the tracker's ability to recover from occlusions. If T is too small, the method will have more difficulty recovering from occlusions. However, a large T may increase chances of linking two different objects.

Following the manifold assumption, we assume that the class labels of the i-th data point and its neighbors can be predicted by a local function $g_i(\cdot)$. Following [19], we adopt the linear regression model as the local prediction function for its simplicity, i.e., $g_i(x_i) = W_i^T x_i + b_i$, where $W_i \in \mathbb{R}^{d \times c}$ is the local projection matrix and $b_i \in \mathbb{R}^c$ is the bias term. A local function only corresponds to a small segment of one trajectory. To exploit the structure of all the trajectories in the entire video sequence, we minimize the prediction error of all the local models g_i for $i = 1, ..., n$, which can be formulated as:

$$\begin{aligned} \min_{W_i|_{i=1}^n, b_i|_{i=1}^n, \tilde{F}_i|_{i=1}^n} \sum_{i=1}^{n} ||X_i^T W_i + \mathbf{1}_{k+1} b_i^T - \tilde{F}_i||_F^2 + \lambda ||W_i||_F^2 \\ s.t. \;\; \tilde{F}_i = [f_i, f_{i_1}, f_{i_2}, \ldots, f_{i_k}]^T. \end{aligned} \tag{4}$$

$||W_i||_F^2$is the regularization term on W_i to control the capacity of W_i. $X_i = [x_i, x_{i_1}, x_{i_2}, \ldots, x_{i_k}] \in \mathbb{R}^{d \times (k+1)}$ comprises the color histograms of the points in the vector N_i^k. $\tilde{F}_i = [f_i, f_{i_1}, f_{i_2}, \ldots, f_{i_k}]^T \in \mathbb{R}^{(k+1) \times c}$ comprises the prediction scores of the points in the vector N_i^k. Denote $H = I - \frac{1}{k+1} \mathbf{1}_{k+1} \mathbf{1}_{k+1}^T \in \mathbb{R}^{(k+1) \times (k+1)}$ as the centering matrix. Following [19], Equation 4 is equivalent to minimizing the following optimization problem:

$$\begin{aligned} \min_F Tr\left(F^T L F\right) \\ s.t. \text{ columns of } F \text{ satisfy Equation 1}, \end{aligned} \tag{5}$$

where L is defined as:

$$L = [S_1, S_2, \ldots, S_n] \begin{bmatrix} L_1 & & \\ & \ldots & \\ & & L_n \end{bmatrix} [S_1, S_2, \ldots, S_n]^T. \tag{6}$$

$L_i \in \mathbb{R}^{(k+1) \times (k+1)}, 1 \leq i \leq n$ is defined as:

$$L_i = H - H X_i^T (X_i H X_i^T + \lambda I)^{-1} X_i H. \tag{7}$$

$S_i \in \mathbb{R}^{n \times (1+k)}, 1 \leq i \leq n$ is a selection matrix which $(S_i)_{pq} = 1$ if $p = (N_i^k)_q$ and $(S_i)_{pq} = 0$ otherwise. L is the Laplacian matrix which encodes all the neighborhood information. For more details on the mathematical derivation, please refer to [19].

3.3. Trajectory Inference

The current objective function shown in Equation 5 is a combinatorial problem, because F is constrained by Equation 1. However, the combinatorial problem is NP-complete. Certain relaxation is required to efficiently solve this objective function. According to Equation 2, F is orthogonal by definition, *i.e.*, $F^T F = I$. Also, all the elements in F is nonnegative by definition, as defined by Equation 1. Furthermore, according to [20], if both the orthogonal constraint and nonnegative constraint are satisfied for a matrix, then there will only be at most one nonzero element in each row of the matrix. If there is at most one nonzero element per row in F, this means that each data point belongs to at most one class, which is exactly the mutual exclusion constraint. Thus we relax the form of F and only keep the orthogonal and nonnegative constraint. In other words, we exploit the orthogonal and *nonnegative* constraints to perform *discretization* of F. Therefore, we propose to minimize Equation 8 to satisfy mutual exclusion and the manifold assumption simultaneously.

$$\min_F Tr\left(F^T L F\right) \;\; s.t. \;\; F^T F = I, F \geq 0. \tag{8}$$

In order to solve Equation 8, we rewrite the objective function as follows:

$$\min_F Tr\left(F^T L F\right) + \tau ||F^T F - I||_F^2 \;\; s.t. \;\; F \geq 0, \tag{9}$$

where τ is a large constant to enforce the orthogonality condition $F^T F = I$. $\tau = 10^{10}$ in our experiments. Following [20], we iteratively optimize Equation 9. The update rule is the following:

- $F_{ij} \leftarrow F_{ij} \frac{(2\tau F)_{ij}}{(LF + 2\tau F F^T F)_{ij}}$
- Normalize F such that $(F^T F)_{ii} = 1$ for $i = 1, \ldots, n$.

The justification of the update rule is detailed in [20].

The main problem of this optimization approach is that the algorithm may converge to a severe local optima. If the initialization is not good, the performance may degrade severely. To have the performance more robust, we resort to weak supervision to get a more stable initialization. More specifically, the initial value of F in our system is obtained by minimizing the following objective function, which takes into account label information.

$$\min_F Tr\left(F^T L F\right) + Tr\left((F - Y)^T U (F - Y)\right), \tag{10}$$

where $U \in \mathbb{R}^{n \times n}$ is a diagonal matrix. $U_{ii} = \infty$ (a large constant) if the i-th data point is a ground truth positive for any class. Otherwise $U_{ii} = 1$. U is used to enforce that the prediction results are consistent with the ground truth positive points. $Y \in \mathbb{R}^{n \times c}$ is the label matrix, where $Y_{ij} = 1$ if the i-th data point is ground truth positive for class j and $Y_{ij} = 0$ if we do not have any label information. According to [19], the global optimal solution for Equation 10 is $F_0 = (L + U)^{-1} U Y$. F_0 is then used as the initial value for solving Equation 9.

The labeled data points in Y are acquired from the manually labeled start and end locations of an individual. For videos with face recognition information available, we can increase the number of labeled data points. If we know by face recognition that the i-th data point is from class j, then we can add this information by setting $Y_{ij} = 1$. By incorporating face recognition, we can prevent the tracker from losing track of the person-of-interest.

Given the predicted label information for each person detection result, the final step is to link together the detections of each individual and infer the trajectory for each individual. A reasonable trajectory should be spatially smooth and follow velocity constraints. Given the start and end location of each individual, a Viterbi search is performed to select the best trajectory for each of the c individuals.

It is worth noting that our formulation can naturally handle template updates, which is crucial because the color of the tracking target can change gradually from time to time. The templates are implicitly encoded in the manifold structure we learn through semi-supervised learning. If the appearance of a tracked object changes smoothly along a manifold, our algorithm can handle the change. In sum, template updating is integrated seamlessly into the entire optimization process.

4. Experiments

We compared our performance with two other trackers on two data sets. The competing algorithms are [3] and a self implemented 3D color particle filter (CPF) similar to [15]. We used HSV color histograms as done in [14]. We split the bounding box horizontally into regions and computed the color histogram for each region similar to the spatial pyramid matching technique [12]. Given L layers, we will have $2^L - 1$ partitions for each template. L is 3 in our experiments. The color particle filter tracks each object independently using a particle filter, which is susceptible to object *hijacking*, i.e. a particle filter starts tracking another particle filter's object. However, the particle filter performs some basic occlusion analysis. If an object is known to be occluded by a wall or some other object, the computation of the color histogram for the occluded object only focuses on the non-occluded part. The starting points of persons-of-interest are given to CPF. For [3], we acquired the source code from the authors and applied it to our data sets. This is an unsupervised method which only relies on background information and does not require any label information. [17] was used for background subtraction. For our method, we used the person detection result from [6, 8]. To describe the person detection result, we used the same kind of color histograms as in CPF. We used the output of the Probabilistic Occupancy Map [7] to filter out all person detection points situated at locations which were deemed to not contain any non-background object. The start and end positions of the persons-of-interest are given to our algorithm. We ran face detection and recognition using the PittPatt software[1].

The evaluation metrics used are the *Multiple Object Tracking Accuracy* (MOTA[2]) and *Precision* (MOTP) from the CLEAR metrics [4], which is the most widely-used evaluation metric to evaluate multi-object tracking algorithms. Following the evaluation method used in [1], association between tracking results and ground truth are computed in 3D with a hit/miss threshold of 1 meter. MOTA takes into account the number of true positives (TP), false positives (FP), missed detections (MD) and identity switches (ID-S). Given the true positive associations, MOTP is the average 3D distance between the ground truth and tracking output.

4.1. Data Sets

We tested our algorithm on two real-world data sets: the PETS 2009 data set and the Caremedia data set. The PETS 2009 [5] S2.L1 sequence consists of 7 cameras and 19 people walking in an open outdoor scene for 795 frames. This is a relatively easy data set because there is one camera (view-001) which has a near global view of the whole scene. All our ground-truths are based on view-001 from [2]. No faces can be extracted due to low resolution of cameras.

The second data set is the Caremedia data set [18] recorded in a nursing home. The surveillance cameras are setup in the public areas. There are many occlusions caused by walls which are typical in indoor scenes. There are also many challenging scenes such as long corridors with sparse camera setups which can easily have much occlusion. There is also no single camera which has a global view of the whole environment, which is typical in many surveillance camera setups, but atypical in the data sets that have been used to perform multi-camera tracking. Furthermore, the data set records activities in a nursing home, where people were focused on their daily tasks and not on the surveillance cameras, which makes the data set a very good representation of situations that may occur in real life. In sum, this is a very challenging real world data set in a complex indoor environment.

The video recordings are 6 minutes 17 seconds long with 11310 frames from 15 cameras. There were a total of 29 trajectories identified, and these trajectories belonged to 13 people in the scene. Patients and staff are all persons-of-interest, because staff at nursing homes interact frequently with the patients, and it is valuable information if we are able to localize and track staff as well. Trajectories that were no more than 3 seconds were ignored. For each trajectory, the ground truth bounding boxes of each tracking target in the 15 cameras are manually labeled at one second intervals. The longest trajectory in the video is 6 minutes 17 seconds, and the shortest trajectory is 4 seconds. The mean length of the trajectories is 39.72 seconds. Faces can be extracted and recognized if a person is close enough to the camera. For our algorithm, we treated each of the 29 trajectories as a class during the learning process. There was a 11310 frame trajectory, which was significantly longer than all the other trajectories. To balance the data set, we split the trajectory into 4 sub-trajectories and concatenated the

[1] Pittsburgh Pattern Recognition (http://www.pittpatt.com)

[2] Code from http://www.micc.unifi.it/lisanti/source-code/.

	MOTA	MOTP	TP	FP	FN	ID-S
[3]	0.684	0.633	3974	1050	426	272
CPF	0.448	0.615	3190	1269	1310	172
Ours-NF	**0.963**	**0.785**	**4565**	**70**	**104**	**3**

(a) Results for PETS 2009 S2.L1 sequence. Ground truth count: 4672.

	MOTA	MOTP	TP	FP	FN	ID-S
[3]	-1.877	0.570	19069	84938	12930	2020
CPF	-0.309	0.583	11772	22527	22014	233
Ours-NF	0.170	0.608	19877	14180	14060	82
Ours-F	**0.762**	**0.632**	**29988**	**4103**	**3983**	**48**

(b) Results for Caremedia sequence. Ground truth count: 34019.

Figure 3. Results for the two evaluation sequences. TP: true positive. FP: false positive. FN: false negative. ID-S: identity switch. *Ours-NF*: our method, no face recognition. *Ours-F*: our method, with face recognition.

sub-trajectories once our learning process was complete.

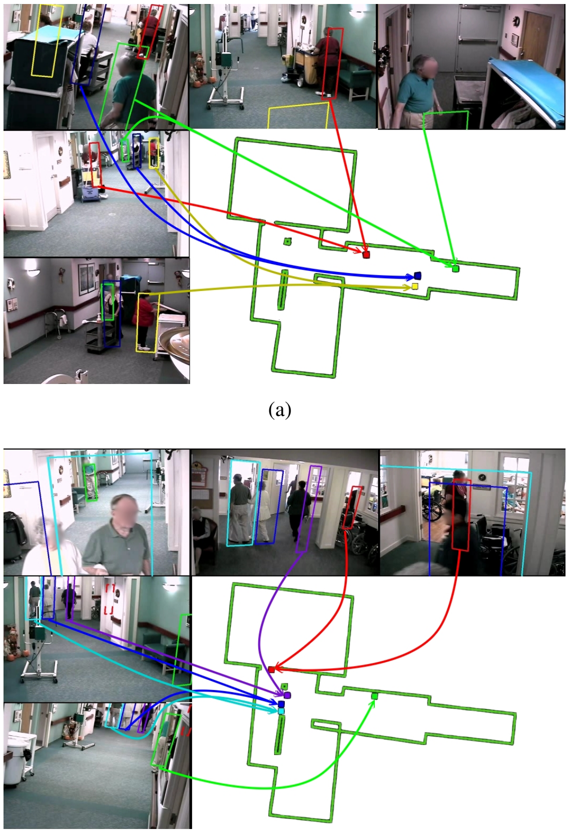
4.2. Results

The results of the PETS 2009 S2.L1 sequence and the Caremedia data set are shown in Table 3(a) and Table 3(b) respectively. Most algorithms perform reasonably well on the relatively easy PETS 2009 S2.L1 sequence. In [3], the authors report 0.77 MOTA for this sequence while we only achieve 0.684. However the authors also mentioned that only 5 out of 7 cameras are used, because there were two cameras with calibration imprecision. We used all 7 cameras, which could be the cause of the performance drop. No face recognition information is available on this sequence, so we ran our method without face recognition (*Ours-NF*).

For the more complex Caremedia data set, the performance gap between different algorithms is much more visible. Although [3] and CPF works fairly well for PETS 2009 S2.L1, they fail in tackling the Caremedia data set. The main reason is that the large number of false positives detected will lower the MOTA score significantly. To understand the impact of face recognition, we run our tracker without face recognition (*Ours-NF*) and with face recognition (*Ours-F*). Our tracker, which utilizes all available information, can still achieve reasonable tracking performance. Snapshots of our algorithm's localization and tracking is shown in Figure 1 and 4. Even though there are 4 cameras viewing the corridor, the clutter in the corridor still causes serious occlusions.

4.3. Discussion: Advantages and Limitations

We analyzed the performance of the compared algorithms. For [3], surveillance camera setups with near-parallel viewing angles of cameras can cause the Probabilistic Occupancy Map (POM) [7] to be inaccurate. For example, cameras setup in a corridor only view the principle direction of the corridor. Therefore, when there is heavy occlusion on the corridor, there can be ambiguity in how to generate a set of person location hypotheses. Also, for locations which are only viewed by one camera, the POM is also not as effective. These are the main difficulties [3] faced. The CPF does not perform well because of two main reasons. The first reason is that the cameras are not color calibrated, which makes cross-camera tracking difficult because the same person may have slightly different color appearance when viewed from different cameras. The second problem is that the CPF does not have the mutual exclusion constraint, and multiple particle filters may start tracking the same object.


(a)

(b)

Figure 4. Snapshots of localization and tracking results from Caremedia data set. To increase readability, not all arrows are drawn.

Our proposed algorithm performs reasonably well due to two main reasons. First, manifold learning coupled with nonnegative discretization is effective in enhancing tracking performance. Our algorithm utilizes the manifold assumption to deal with slight color differences of the tracking target at different times. The nonnegative discretization enforces the mutual exclusion constraint. Second, we utilize PittPatt face recognition, which is a very reliable source of label information. Face recognition provides more labels for each tracking target. Face recognition also helps overcome the color-mismatch problem the CPF faces, because if a face is detected in a given camera, then the color histogram of the person for the camera is known. Without faces, manifold learning with nonnegative discretization already achieves scores which are already better than our baselines on both data sets. By incorporating face recognition (*Ours-F*), we further improve the MOTA score of the Caremedia data set to 0.762. Even though our algorithm requires the start and end information of a track, we emphasize that labeling the additional information requires very little human labor, but can lead to substantial improvement in accuracy. In sum, experiments show that by incorporating all available cues and exploiting the manifold assumption with nonnegative discretization, we can achieve substantial improvements in performance.

There are still limitations to our algorithm. First, our objective function does not have a spatial locality constraint on a trajectory, *i.e.*, an individual cannot be at multiple places at the same time. Therefore, our algorithm is not effective in very crowded sequences where each person wears the same color clothes, such as the *laboratory* sequence from [3]. Second, the optimization may converge to a severe local optima, which makes the initialization very important. Bad initialization may cause the performance to degrade. We plan to solve these issues in the future.

5. Conclusions

We propose a novel semi-supervised learning framework with nonnegative discretization to incorporate all available cues to perform robust person-of-interest localization and tracking in complex indoor environments. Available cues such as color, person detection, face recognition and non-background information are all utilized in the manifold learning process. The nonnegative discretization groups the data points into non-overlapping groups such that mutual exclusion and manifold assumption are satisfied simultaneously. We have shown in our experiments that our method is effective in both outdoor and complex indoor environments. Our algorithm is effective because of reliable face recognition and the combination of manifold learning with nonnegative discretization.

Acknowledgements

This work is supported in part by the National Science Foundation under Grants IIS-0917072. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] A. Andriyenko, K. Schindler, and S. Roth. Discrete-continuous optimization for multi-target tracking. In *CVPR*, 2012. 1, 2, 5

[2] C. Beleznai, D. Schreiber, and M. Rauter. Pedestrian detection using gpu-accelerated multiple cue computation. In *CVPR Workshops*, 2011. 5

[3] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua. Multiple object tracking using k-shortest paths optimization. In *IEEE TPAMI*, 2011. 1, 2, 5, 6, 7

[4] K. Bernardin and R. Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. In *J. Image Video Process.*, 2008. 5

[5] A. Ellis, A. Shahrokni, and J. Ferryman. Pets2009 and winter-pets 2009 results: A combined evaluation. In *Performance Evaluation of Tracking and Surveillance (PETS-Winter)*, 2009. 5

[6] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. In *IEEE TPAMI*, 2010. 5

[7] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua. Multicamera People Tracking with a Probabilistic Occupancy Map. In *IEEE TPAMI*, 2008. 5, 6

[8] R. B. Girshick, P. F. Felzenszwalb, and D. McAllester. Discriminatively trained deformable part models, release 5. http://people.cs.uchicago.edu/ rbg/latent-release5/. 5

[9] H. Grabner, C. Leistner, and H. Bischof. Semi-supervised on-line boosting for robust tracking. In *ECCV*, 2008. 2

[10] H. Jiang, S. Fels, and J. J. Little. A linear programming approach for multiple object tracking. In *CVPR*, 2007. 2

[11] S. M. Khan and M. Shah. Tracking Multiple Occluding People by Localizing on Multiple Scene Planes. In *IEEE TPAMI*, 2009. 1, 2

[12] S. Lazebnik, C. Schmid, and J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In *CVPR*, 2006. 5

[13] A. Mittal and L. S. Davis. M2Tracker: A Multi-View Approach to Segmenting and Tracking People in a Cluttered Scene. In *IJCV*, 2003. 2

[14] K. Okuma, A. Taleghani, N. D. Freitas, O. D. Freitas, J. J. Little, and D. G. Lowe. A boosted particle filter: Multitarget detection and tracking. In *ECCV*, 2004. 1, 2, 5

[15] P. Perez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. In *ECCV*, 2002. 5

[16] S. Stalder, H. Grabner, and L. V. Gool. Beyond semi-supervised tracking: Tracking should be as simple as detection, but not simpler than recognition. In *3rd On-line learning for Computer Vision Workshop*, 2009. 2

[17] C. Stauffer and W. Grimson. Adaptive background mixture models for real-time tracking. In *CVPR*, 1999. 5

[18] Y. Yang, A. Hauptmann, M.-Y. Chen, Y. Cai, A. Bharucha, and H. Wactlar. Learning to predict health status of geriatric patients from observational data. In *Computational Intelligence in Bioinformatics and Computational Biology*, 2012. 5

[19] Y. Yang, F. Nie, D. Xu, J. Luo, Y. Zhuang, and Y. Pan. A multimedia retrieval framework based on semi-supervised ranking and relevance feedback. In *IEEE TPAMI*, 2012. 2, 3, 4

[20] Y. Yang, H. T. Shen, F. Nie, R. Ji, and X. Zhou. Nonnegative spectral clustering with discriminative regularization. In *AAAI*, 2011. 4