An Algorithm for Multiple Object Trajectory Tracking

Mei Han Wei Xu Hai Tao[‡] Yihong Gong
NEC Laboratories America, Cupertino, CA, USA
{meihan, xw, ygong}@sv.nec-labs.com
[‡]University of California at Santa Cruz, Santa Cruz, CA, USA
tao@soe.ucsc.edu

Abstract

Most tracking algorithms are based on the maximum a posteriori (MAP) solution of a probabilistic framework called Hidden Markov Model, where the distribution of the object state at current time instance is estimated based on current and previous observations. However, this approach is prone to errors caused by temporal distractions such as occlusion, background clutter and multi-object confusion. In this paper we propose a multiple object tracking algorithm that seeks the optimal state sequence which maximizes the joint state-observation probability. We name this algorithm trajectory tracking since it estimates the state sequence or "trajectory" instead of the current state. The algorithm is capable of tracking multiple objects whose number is unknown and varies during tracking.

We introduce an observation model which is composed of the original image, the foreground mask given by background subtraction and the object detection map generated by an object detector. The image provides the object appearance information. The foreground mask enables the likelihood computation to consider the multi-object configuration in its entirety. The detection map consists of pixelwise object detection scores, which drives the tracking algorithm to perform joint inference on both the number of objects and their configurations efficiently.

1. Introduction

Multiple object tracking has been a challenging topic in computer vision. Generally speaking, it has to solve two problems jointly: the estimation problem as in traditional tracking, and the data-association, especially when multi-object interaction exists. Many tracking algorithms solve the estimation problem in a maximum *a posteriori* (MAP) formulation [2], that is, the tracking result is the current object state with the largest posterior probability based on current and previous observations. The MAP formulation can be simplified if the Hidden Markov Model (HMM) [12] is assumed. Estimation algorithms that find the MAP solution or propagate this distribution over time are equivalent to the forward algorithm in HMM literature. However, this approach may fail with background clutter, occlusion and multi-object confusion. Another type of tracking algorithms estimates the joint state-observation sequence distribution. The tracking result corresponds to the *state sequence* which maximizes the joint probability between the state sequence and the observation sequence. The state sequence indicates the trajectories of the multiple objects being tracked. We call this type of tracking method *trajectory tracking*.

In a discrete HMM model, the trajectory tracking problem can be solved using the Viterbi algorithm [12], which is a dynamic programming algorithm that keeps all best sequences ending at all possible states in current frame. A well-known early work in trajectory tracking is the multiple hypothesis tracking (MHT) algorithm developed by Reid [13]. In MHT, the multi-target tracking problem is decomposed into the state estimation and the data association components. The data association problem is NP-hard though some work has been done to generate k-best hypotheses in polynomial time [3]. The joint probabilistic data association filter (JPDAF) [4] finds the state estimate by evaluating the measurement-to-track association probabilities. Some methods are presented to model the data association as random variables which are estimated jointly with state estimation by EM iterations [15, 5]. Most of these methods are in the small target tracking community where object representation is simple.

There has been much work on multiple object visual tracking. MacCormick and Blake use a sampling algorithm for tracking a fixed number of objects [11]. Tao et al. present an efficient hierarchical algorithm to approximate the inference process in high dimensional space [16]. Isard and MacCormick propose a Bayesian multiple-blob tracker and use a particle filter to perform the inference [9]. Hue et al. describe an extension of the classical particle filter where the stochastic assignment vector is estimated by a Gibbs sampler [8]. These tracking methods do not perform trajectory tracking mostly due to the computation cost. In theory, a tracking algorithm should pursue the best joint state-observation sequence instead of the marginal

1063-6919/04 $20.00 © 2004 IEEE

distribution of the current state. However, when the state dynamics and likelihood are single Gaussian distributions, the joint state-observation is also a single Gaussian distribution. In this case, the MAP solutions of the joint distribution and the marginal distribution have identical current state. Therefore, there is no advantage in using a trajectory tracking scheme. For general distributions, however, the marginal tracker is no longer a good approximation to the trajectory tracker. This phenomenon is more prominent for multi-object tracking. Another reason why the trajectory tracking is not widely used is that the state variables are mostly continuous variables such as positions, deformation parameters and appearance values. Even though these variables can be quantized, the number of all possible combinations is often very large. The situation becomes even worse when multiple objects are tracked jointly.

We introduce a novel observation model for efficient multiple object trajectory tracking. The observation is composed of the image itself, the foreground mask given by a background modelling method and the object detection map generated by an object detector. The image provides the object appearance information which helps to keep the tracking identity even with multi-object interaction and occlusion. A Gaussian-mixture based adaptive background modelling [14] is used to generate a binary foreground mask image. This mask image enables the likelihood computation to consider the multi-object configuration in its entirety. The detection map consists of pixel-wise object detection scores. Any object detection method can be used. In our implementation, we apply a convolutional neural network based object detection module [10] to detect pedestrians. The detection score map makes the inference process work in a limited discrete space, which dramatically improves the computation efficiency.

Many people have worked on the integration of object detection and tracking. SVM tracker applies recognition algorithms to efficient visual tracking [1, 18]. Many systems of multiple people detection and tracking are presented using aspect ratio [6], silhouette [7], human shape model [19] to detect human. Verma et al. describe a method to propagate the probabilities of face detections over time [17].

In this paper, we present an algorithm for multiple object trajectory tracking. It can deal with temporal occlusions, background clutter and multi-object interactions. The number of objects may vary during tracking. A novel observation model is introduced to take advantage of visual information, consider multiple objects jointly and most importantly, perform an efficient inference process driven by the object detection score map. Experimental results on human tracking are presented and applications to traffic control and anomaly detection are described.

2. Observation

The observation is composed of the original image frame, the foreground mask image given by background modelling and the object detection map generated by an object detector, as shown in Figure 1.

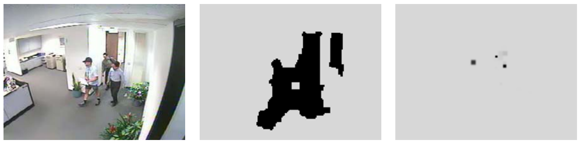
A Gaussian-mixture based adaptive background modelling method [14] is used to generate a binary foreground mask image as shown in Figure 1 (b). The black pixels represent the mask of the foreground objects. This image serves two purposes. Firstly, it enables the likelihood computation to consider the multi-object configuration in its entirety. It gives a measure about how well a configuration, including the number of objects and their states, interprets the foreground pixels, which is described in Section 3. It takes advantage of the visual information in the likelihood computation. Secondly, we can restrict the object detector to search over the foreground areas only where the probability of detecting an object is high. It reduces the amount of processing required.

The detection map, shown in Figure 1(c), consists of pixel-wise object detection scores. Any object detection method can be used. In our implementation, we apply a convolutional neural network based object detection module [10] to detect pedestrians. Each foreground blob is potentially the image of a person. Each pixel location is applied to the trained neural network that identifies human forms. The neural network generates a score, or probability, indicative of the probability that the blob around the pixel does in fact represent a human of some scale. A particular part of the detected person, e.g., the approximate center of the top of the head, is illustratively used as the "location" of the object, which is shown as a dark spot in Figure 1(c). The darker spot demonstrates the higher detection score. The neural network searches over each pixel at a few scales. The detection score corresponds to the best score, i.e., the largest detection probability, among all scales. The detection score map makes the inference process work in a limited discrete space which dramatically improves the computation efficiency.

The image provides the object appearance information which helps to keep the tracking identity. For each location in the image, the object detector provides a bounding box whose size corresponds to the scale given by the best detection score at that location. The object appearance at the location is represented by the color histogram calculated within the bounding box.

3. Multiple Object Trajectory Tracking

The multiple object trajectory tracking method uses a Hidden Markov Model to get the best joint probability of state



(a) (b) (c)

Figure 1: Observation: (a) original image, (b) foreground mask image, the black pixels represent the mask of the foreground objects, (c) human detection score map, the darker pixels show the higher detection scores.

sequence and observation sequence. The hidden state includes information about the location, velocity, appearance and scale of each object. Each object is identified by index i and its state at time t is represented by

$$x_t^i = (p_t^i, v_t^i, a_t^i, s_t^i) \tag{1}$$

where p_t^i is the image location, v_t^i represents the 2D velocity, a_t^i and s_t^i denote the appearance and scale of object i at time t, respectively. p_t^i and v_t^i use continuous image coordinates. We use the color histogram a_t^i to represent the object appearance, and the size of the bounding box within which the color histogram is computed to represent the scale s_t^i, as described in Section 2. We describe all the objects in the image using a multi-object configuration, $X_t = \{x_t^i | i = 1 \cdots K\}$, which is the state at time t. It is a set of object parameters. K is the number of objects at time t. If M is the maximum possible number of objects in an image, the configuration space is $\bigcup_{K=0}^{M} X_t^K$, representing all possible configurations with 0 to M objects where X_t^K denotes the configuration space composed of K objects. For trajectory tracking, a configuration sequence that maximizes $P(X_t, \cdots, X_0 | Z_t, \cdots, Z_0)$ is pursued. It should be noticed that the superscript K is ignored because the number of objects may change over time. Z_t is the observation at time t, which is composed of the image Z_{It}, the foreground mask Z_{Ft} and the detection map Z_{Dt}.

The joint probability $P(Z, X)$ of given state sequence X and observation sequence Z can be written as,

$$P(Z, X) = P(Z|X)P(X) \tag{2}$$

where

$$\begin{aligned} P(X) &= P(X_0) \prod_{t=1} P(X_t | X_{t-1} \cdots X_0) \\ P(Z|X) &= P(Z_0) \prod_{t=1} P(Z_t | X, Z_{t-1}, \cdots, Z_0) \end{aligned} \tag{3}$$

With the assumption of Markov property, we have,

$$\begin{aligned} P(X_t | X_{t-1} \cdots X_0) &\approx P(X_t | X_{t-1}) \\ P(Z_t | X, Z_{t-1}, \cdots, Z_0) &\approx P(Z_t | X_t) \end{aligned} \tag{4}$$

3.1. Multiple Object Dynamics

Let $A(t)$ be the set of the indices of the active objects, i.e., the objects appear in the view, at time t. Assuming independent movement of each object, $P(X_t | X_{t-1})$ can be factorized as,

$$P(X_t | X_{t-1}) \propto \tau_1 \, \tau_2 \, \tau_3 \tag{5}$$

where

$$\begin{aligned} \tau_1 &= \prod_{i \in A(t) \bigcap A(t-1)} P(x_t^i | x_{t-1}^i) \\ \tau_2 &= \prod_{i \in A(t) \setminus A(t-1)} P_a(p_t^i) \\ \tau_3 &= \prod_{i \in A(t-1) \setminus A(t)} P_a(p_{t-1}^i) \end{aligned} \tag{6}$$

The first term τ_1 represents the state transition dynamics,

$$x_t^i = \Phi x_{t-1}^i + \Sigma w_{t-1} \tag{7}$$

where Φ is the state transition matrix, Σ is the disturbance matrix and w_{t-1} is normal random noise with zero mean and covariance Q. We employ a constant velocity model, therefore, Φ is defined as,

$$\Phi = \begin{bmatrix} I_{2\times2} & I_{2\times2} & 0 & 0 \\ 0 & I_{2\times2} & 0 & 0 \\ 0 & 0 & I_{m\times m} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

where $I_{2\times2}$ and $I_{m\times m}$ denote 2-by-2 and m-by-m identity matrices, m is the number of quantization bins in the color histogram representation.

The second term τ_2 models object addition or new track initialization and the third term τ_3 for object deletion or track ending. The function P_a gives the probability that a trajectory does, in fact, begin (appear) or end (disappear) in the current frame. P_a is a function of the object location p_t^i. The probability is higher at the edges of the field and the door area than in the center because people typically do not appear or disappear "out of nowhere", i.e., in the middle of the field [16].

3.2. Observation Likelihood

The probability $P(Z_t | X_t)$ describes how the underline state X_t of the system fits the observation Z_t. An object-based likelihood function is often computed as the matching score of the object representation with the image at the object location. Such a likelihood function does not explain the whole image. On the other hand, an image-based likelihood function explains every pixel in the image using the object state. The advantage of using the image-based likelihood is that when the tracker is trapped at a wrong location, such as a background clutter, the likelihood is low because the true target cannot be explained using other objects. We propose a combination likelihood computation which is composed of an object-based likelihood computation for the original image, and image-based likelihood computations for the foreground mask and the detection map.

Assuming the three observations are independent given the state,

$$P(Z_t|X_t) = P(Z_{It}|X_t)\, P(Z_{Ft}|X_t)\, P(Z_{Dt}|X_t) \qquad (9)$$

where $P(Z_{It}|X_t)$ represents the image observation likelihood, $P(Z_{Ft}|X_t)$ denotes the foreground mask likelihood and $P(Z_{Dt}|X_t)$ is the detection score map likelihood. We get the object size and appearance from the original image according to the position of the object p_t^i and the detection map Z_{Dt} at time t . The size I_s^i, which is a function of p_t^i and Z_{Dt}, is the scale with the best human detection score at the closest discrete position to p_t^i because human detector only works on pixel-wise coordinates. The appearance from the image I_a^i, which is a function of p_t^i and I_s^i, is computed within the bounding box at p_t^i of size I_s^i. Therefore,

$$P(Z_{It}|X_t) = \prod_i P(I_a^i|a_t^i) P(I_s^i|s_t^i) \qquad (10)$$

where $P(I_a^i|a_t^i)$ and $P(I_s^i|s_t^i)$ have Gaussian distributions $N(a_t^i, \Sigma)$ and $N(s_t^i, \sigma)$.

$P(Z_{Ft}|X_t)$ is the likelihood of the foreground mask, which is composed of coverage and compactness, as in [16]. The main idea is to use background subtraction result as the cue for detecting the foreground region. The areas without significant image changes are considered to be well explained by the background model. The object state then only needs to explain all the changing blobs. The likelihood is computed as the percentage of the covered changing blobs, or the coverage, and the efficiency of multi-object coverage, or the compactness.

The object detector calculates the score of the object presence at location j, where j is a discrete coordinate. We assume the following noisy detection process. An object at position p is detected at location j with probability $f(j|p)$. A single object can cause object detection responses at multiple locations and the detection response at a single location can be caused by multiple objects. The response of object detection at location j depends on a binary random variable H_j indicating whether Z_{Dt}^j is a response caused by some object(s). Further more, we assume that the detection response at different locations are independent with each other. We have,

$$\begin{aligned} P(Z_{Dt}|X_t) &\propto \prod_j P(Z_{Dt}^j|X_t) \\ &= \prod_j \sum_{H_j} P(Z_{Dt}^j|H_j) P(H_j|X_t) \quad (11) \end{aligned}$$

and

$$\begin{aligned} P(H_j = 0|X_t) &= \prod_i (1 - f(j|p_t^i)) \\ P(H_j = 1|X_t) &= 1 - P(H_j = 0|X_t) \qquad (12) \end{aligned}$$

The distributions $P(Z_{Dt}^j|H_j)$ and $f(j|p_t^i)$ encode the prior knowledge about how the observation relates to the internal state of the system.

3.3. Detection and Tracking

The configuration space of multiple object tracking has a very high dimension. The key to the success of trajectory tracking for multi-object configuration is to develop an efficient inference algorithm in the configuration space. In the multiple hypothesis tracking (MHT) algorithm [13, 3], this problem is solved for small targets by finding all possible combinations of current observations and existing trajectories within certain clusters. We propose a detection-driven inference algorithm which is a combination of state estimation and data association. We make the process more efficient by local pruning based on the likelihood computation.

The detection score map is used to quantize the configuration space, i.e., we treat the object detections in a similar way as the multiple hypothesis tracking algorithms deal with radar signals. We only work on object detections at discrete locations where the detection score is the local maximum and larger than some threshold. In some sense, the state dynamics and the image likelihood $P(Z_{It}|X_t)$ are performing the data association in MHT. The detection based observation likelihood $P(Z_{Dt}|X_t)$ and the foreground likelihood $P(Z_{Ft}|X_t)$ both perform image-based evaluation of multi-object configurations. Together with the data association value, the multiple object trajectory tracking prefers the tracks which are connections of object detections with high detection scores, undergo smooth movements and have better interpretations of the foreground blobs. The difference to small target tracking is that we make use of the visual information. The image likelihood takes advantage of the appearance information from the original image. The foreground based observation likelihood considers the multi-object configuration simultaneously. The strong visual cues make the configuration sequences with the best joint state-observation probability survive temporal missing/false detections, occlusions and cluttered background.

We also enforce a maximum number of tracks which are active in the view simultaneously. To further improve the computation efficiency, we perform local pruning using either a threshold for likelihood value or a maximum number of configurations kept in the inference. With reasonable assumptions of the thresholds, we can achieve real-time performance in a not-too-crowded environment. These thresholds certainly have the risk of eliminating correct sequences, especially when the measurements are weak. Our empirical results show that our pruning technique gives optimal or close-to-optimal tracking results with reasonable performance of background modelling and object detection. Design of an optimal inference algorithm for multiple object trajectory tracking with tractable computation cost remains an open research problem.

4. Experiment

The multiple object trajectory tracking method has been tested on two existing CCTV cameras which were installed to monitor entries with access controls. These two environments are semi-outdoor. Since object detection is included in the observation sequence, the requirement for accurate background modelling is relaxed and the method can also be applied to outdoor environment. The advantage of trajectory tracking is that the statistical model is searching for the best joint probability between the state sequence and the observation sequence. It makes a temporally global decision, therefore, it allows errors in background modelling, and missing/false detections from object detector.

4.1. Tracking Results

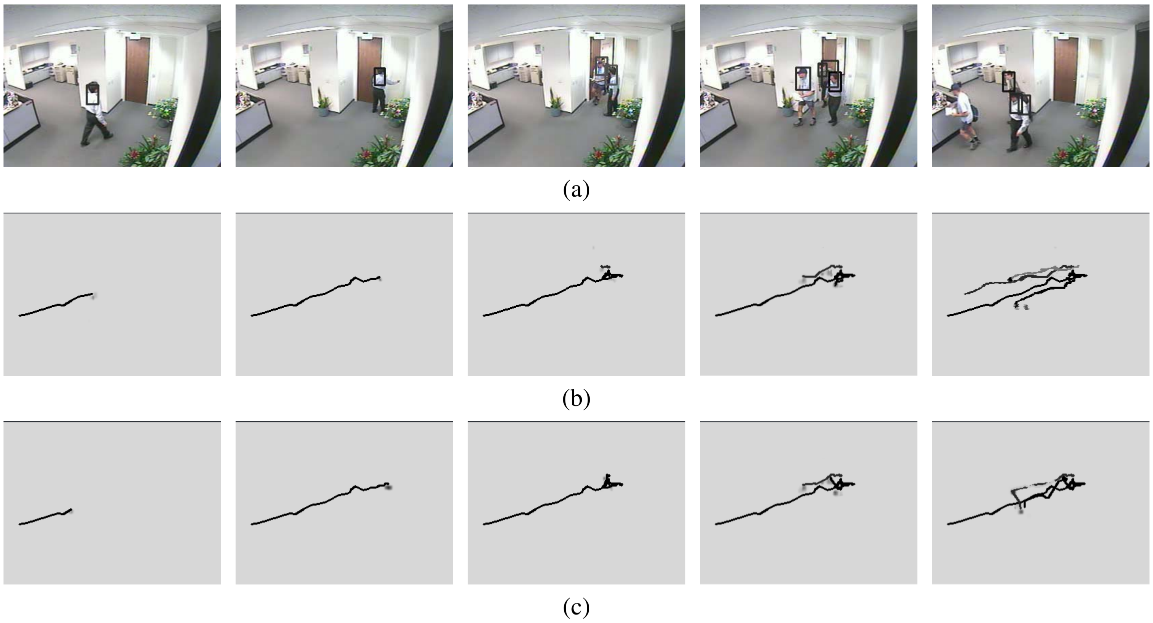
In this section we describe our experimental results of multiple object trajectory tracking. The first example demonstrates the importance of trajectory tracking, especially when the human detection makes mistakes. The next two examples show how the observations, i.e., the image, the foreground mask and the detection map, help to achieve good tracking results.

The first scenario is taken inside the secure area of an access control door. Three persons are involved. Person A first goes up to the door and opens the secure door from inside for persons B and C to come in. Then three of them walk inward. Figure 2(a) shows 5 images from the sequence with overlaid bounding boxes indicating the human detection results. The darker bounding box demonstrates the higher detection score and the size of the bounding box represents the scale with the best detection score. Figure 2(b) demonstrates the multi-tracks with the largest joint state-observation sequence probability generated by the multiple object trajectory tracking. The tracks are overlaid on the detection score map. Different intensities represent different tracks. Due to lighting changes and the semi-outdoor environment, the foreground mask sometimes is not complete and/or has false masks on background clutter. The human detection based on each image is certainly not perfect either. There are false detections in the forth and fifth images caused by background noise and people interaction. In the third image, the human detector misses the person in the back due to occlusions and in the fifth image, it misses the person in the front due to distortions. However, the trajectory tracking manages to maintain the right number of tracks and their configurations, as shown in Figure 2(b), because it searches for the best explanation sequence of the observations over time. The fifth image shows three tracks, the turning track at the bottom (the darkest one) corresponds to person A who walks to the door and comes back, the track in the middle indicates the person wearing a hat and the short track on the top (the lightest one) represents the last person coming in. The first four images in Figure 2(b) show the corresponding multi-track or multi-object configurations at the moments when the images in (a) are taken. Therefore, we know when and where each track begins and the interactions among the tracks.

Since we are performing trajectory tracking, a lot of other object configuration sequences exist. We show one of them in Figure 2(c) which has a lower joint probability at the end of the sequence. This sequence starts with one object track, as shown in the first two images, which is quite similar to the optimal solution. Therefore, the joint probability is also very high up to the moment when more people appear. In the third image, the second person appears and the configuration chooses to maintain one track for both detections. The detection-based observation likelihood, at this moment, is lower than the optimal one in Figure 2(b) where a new track is initialized for the person who just walks in. In the forth and fifth images, the configuration only keeps two tracks to interpret the observations. Due to occlusions between the three people, there have been quite a few missing detections as well as false detections. Therefore, this configuration sequence generates two jumpy tracks which are switching between different people to cover the shaky detections. However, the foreground mask likelihood and the image likelihood, which is based on appearance and size matching, are both lower than the optimal one.

Figure 3 demonstrates an example of multiple people tracking at another access control area. This camera is mounted at a non-secure area where the card swiping machine can be seen. The example first shows the lady opens the door for the person in gray shirt, then the person in dark shirt follows and goes into the area. Figure 3(a) shows the images from the sequence and (b) demonstrates the tracking results. Since people stop and talk in the sequence, the state dynamics cannot guarantee to maintain the correct track identities when people cross each other. The image likelihood composed of appearance and size, however, is able to maintain the track identities, as shown in Figure 3(b). Interestingly, there is one short track close to the up-left corner of the images because one person is standing inside the secure area and the human detection consistently detects him through the glass window. Therefore, 4 tracks are shown in Figure 3(b), the short track for the standing person, the long track for the lady, the dark track for the guy in gray shirt, and the light track for the guy in dark shirt.

The last example demonstrates that the object detec-



(a)

(b)

(c)

Figure 2: Trajectory tracking results with missing/false human detections: (a) original images with overlaid bounding boxes showing the human detection results, (b) the tracking results with the best joint state-observation probability (three tracks are included), (c) the tracking results with a lower joint probability (only two tracks are maintained).

tion not only improves the computation efficiency, but also makes the tracking system have the "recognition" capability. Figure 4 shows a cleaning lady walking into the door, loitering around a plant and then walking away. It is difficult to get the outline of the people from the foreground mask by contour matching [7, 19] or aspect ratio [6] because of occlusions and the existence of other large objects, such as the trash can in this example. Even the foreground mask likelihood computation can be fooled because the foreground pixels of the large object cannot be covered by the human. However, the human detector provides the information to distinguish the person and the irrelevant object. Our trajectory tracking takes advantage of the neural network human detector so that it "recognizes" the human and maintains one clean track.

In these experiments, we use the detection-driven inference algorithm presented in Section 3.3. We assume that the maximum number of people in the scene is 6 and keep the 30-best configurations. With these numbers set, we achieve real-time performance at 10 fps including background modelling, human detection and trajectory tracking.

4.2. Anomaly Detection

The trajectory tracking results provide information accumulated through image frames, including the number of objects, their motion history and interaction, the timing of their behaviors. In practice, the information can only be extracted from a vision system. For example, it is possible to set up a sensor for indicating the opening and closing of the door. However, without a camera system, there is no way to confirm if objects (people, vehicles, animals, luggage) are actually through the door and how many are through the same door around the same time. Basically the multiple object visual tracking system interprets the image sequences about WHO (how many), WHEN, WHERE and WHAT. Based on the information provided by the tracking results which consist of the number of objects and the corresponding multiple tracks, we can analyze the behaviors of the objects. The goal is to detect abnormal behaviors which are related to object counting, interaction, motion and timing. Since objects are tracked all the time, we can also provide a scheme to report traffic information: how many objects are getting in/out the door and loiter/stay at sensitive areas at certain time.

We have tested the trajectory tracking system to detect tailgating and piggy-backing violations at access points. If there are more than one object going into secure area with only one card swipe, it is a tailgating violation. If the object going into secure area is not the object who just swipes the card, it is piggy-backing. We also define rules to detect suspicious behaviors, such as multiple swipes by one object, object loitering at door or swiping areas. It is straightforward to integrate the signals from card reader, door opening and other sensors into the system. As mentioned above,

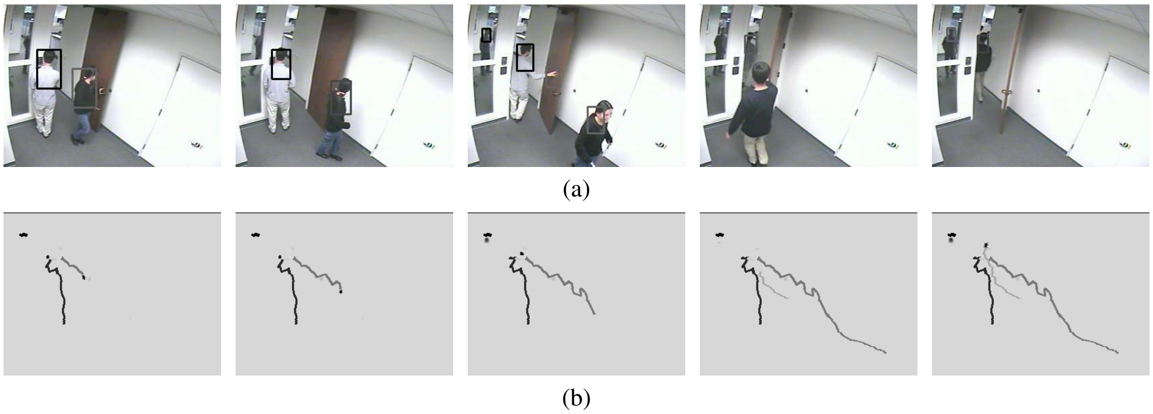


Figure 3: Trajectory tracking results of crossing tracks: (a) original images with overlaid bounding boxes showing the human detection results, (b) multiple object tracking results with correct track identities.

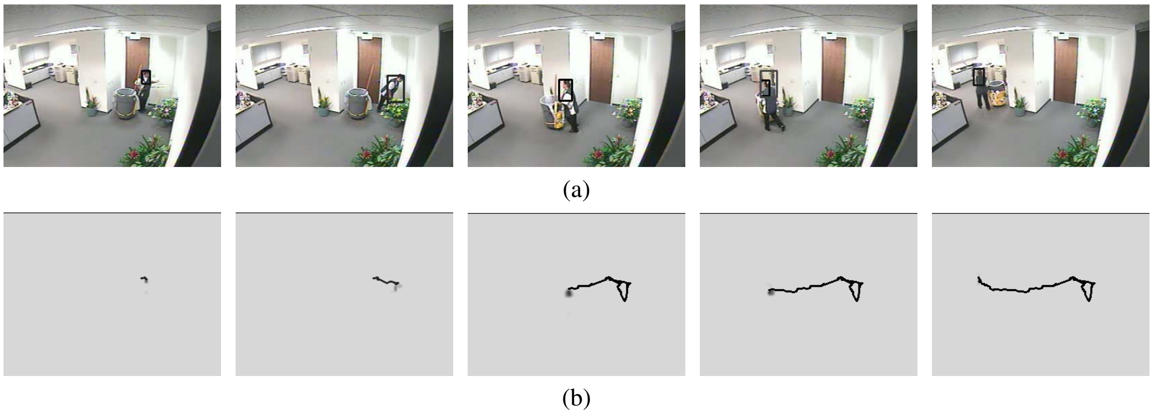


Figure 4: Trajectory tracking results with large objects: (a) original images with overlaid bounding boxes showing the human detection results, (b) multiple object tracking results showing a track which only involves the person.

videos	events	violations	detected violations	false alarms
access door 1	772	37	35	4
access door 2	943	45	42	5

Table 1: Recall and precision of violation detection on one month's real videos for two access control doors.

even with these signals we still need images and robust tracking to verify if objects are at door or swiping areas, when they appear/disappear, how they have been moving and when. In the cases when we do not have door opening or card swiping information, we can get the related information from images.

The system has been running at a few access control areas, where only the visual information is available, for more than six months. In average, the system achieves 98.3% precision in events classification. The violation detection rate is 90.4% and the detection precision is 85.2%. Table 1 lists the results of one month's data at two access doors. The system gets overall 99.2% precision. The violation recall and precision rates are 93.9% and 89.5%, respectively. Details are shown in Table 1. The results are very satisfying.

5. Conclusion

In this paper we describe a multiple object trajectory tracking method. We use a Hidden Markov Model as the probabilistic model to maximize the joint probability between the state sequence and the observation sequence. Therefore, the trajectory tracking method can deal with temporally local difficulties generated by cluttered background, multi-object interaction and occlusions. We use an observation model composed of the original image, the foreground mask after background subtraction and the object detection score map generated by an object detector. Neither the background modelling nor the object detection is required to be accurate. Both background modelling and object detection are making image-based decisions while the trajectory tracking is making a global decision on the number of objects and their configurations. We show a detection-driven inference scheme to perform reliable and efficient multi-object tracking without requirement on the number of objects to be known or fixed.

The configuration space of multiple object trajectory tracking has a high dimension. Much more work has to be done to design an efficient inference algorithm. Learning of the model parameters is another challenging task. Our future work includes the improvement of the current inference algorithm and the development of learning algorithms for the HMM described in this paper.

References

[1] S. Avidan. Support vector tracking. In *CVPR01*, pages I:184–191, 2001.

[2] Y. Bar-Shalom and X.R. Li. *Multitarget Multisensor Tracking: Principles and Techniques*. YBS Publishing, 1995.

[3] I.J. Cox and S.L. Hingorani. An efficient implementation of reid's multiple hypotheses tracking algorithm and its evaluation for the purpose of visual tracking. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(2):138–150, Feb 1996.

[4] T.E. Fortmann, Y. Bar-Shalom, and M. Scheffe. Sonar tracking of multiple targets using joint probabilistic data association. *IEEE Journal Oceanic Eng.*, OE-8:173–184, July 1983.

[5] H. Gauvrit and J.P. Le Cadre. A formulation of multitarget tracking as an incomplete data problem. *IEEE Trans. on Aerospace and Electronic Systems*, 33(4):1242–1257, Oct 1997.

[6] I. Haritaoglu, D. Harwood, and L.S. Davis. W4s: A realtime system for detecting and tracking people in 2 1/2-d. In *ECCV98*, 1998.

[7] I. Haritaoglu, D. Harwood, and L.S. Davis. Hydra: Multiple people detection and tracking using silhouettes. In *VS99*, 1999.

[8] C. Hue, J.P. Le Cadre, and P. Perez. Tracking multiple objects with particle filtering. *IEEE Trans. on Aerospace and Electronic Systems*, 38(3):791–812, July 2002.

[9] M. Isard and J.P. MacCormick. Bramble: A bayesian multiple-blob tracker. In *ICCV01*, pages II: 34–41, 2001.

[10] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, November 1998.

[11] J.P. MacCormick and A. Blake. A probabilistic exclusion principle for tracking multiple objects. In *ICCV99*, pages 572–578, 1999.

[12] L.R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:257–286, Feb 1989.

[13] D.B. Reid. An algorithm for tracking multiple targets. *AC*, 24(6):843–854, December 1979.

[14] C. Stauffer and W.E.L. Grimson. Learning patterns of activity using real-time tracking. *PAMI*, 22(8):747–757, August 2000.

[15] R.L. Streit and T.E. Luginbuhl. Maximum likelihood method for probabilistic multi-hypothesis tracking. In *Proceedings of SPIE International Symposium, Signal and Data Processing of Small Targets*, 1994.

[16] H. Tao, H.S. Sawhney, and R. Kumar. A sampling algorithm for tracking multiple objects. In *Vision Algorithms 99*, 1999.

[17] R.C. Verma, C. Schmid, and K. Mikolajczyk. Face detection and tracking in a video by propagating detection probabilities. *PAMI*, 25(10):1215–1228, October 2003.

[18] O. Williams, A. Blake, and R. Cipolla. A sparse probabilistic learning algorithm for real-time tracking. In *ICCV03*, pages 353–360, 2003.

[19] T. Zhao, R. Nevatia, and F. Lv. Segmentation and tracking of multiple humans in complex situations. In *CVPR01*, pages II:194–201, 2001.