Tracking Interacting Targets with Laser Scanner via On-line Supervised Learning

Xuan Song, Jinshi Cui, Xulei Wang, Huijing Zhao and Hongbin Zha
State Key Laboratory of Machine Perception, Peking University, China
{songxuan,cjs,wangxulei,zhaohj,zha}@cis.pku.edu.cn

Abstract— Successful multi-target tracking requires locating the targets and labeling their identities. For the laser based tracking system, the latter becomes significantly more challenging when the targets frequently interact with each other. This paper presents a novel on-line supervised learning based method for tracking interacting targets with laser scanner. When the targets do not interact with each other, we collect samples and train a classifier for each target. When the targets are in close proximity, we use these classifiers to assist in tracking. Different evaluations demonstrate that this method has a better tracking performance than previous methods when interactions occur, and can maintain correct tracking under various complex tracking situations.

I. INTRODUCTION

Multiple targets tracking plays an important role in various applications, such as surveillance, human motion analysis, traffic flow analysis and others. Multi-target tracking is much easier when the targets are distinctive or not interacting with each other because it can be solved by using multiple independent trackers. However, for the targets similar in appearance, obtaining the correct trajectories of them becomes significantly more challenging when they interact. Specifically, for the laser-based human tracking systems [1-3] (as shown in Fig. 1), which cannot provide the color information of targets, maintaining the correct tracking seems to be an impossible mission when the well-known “merge/split” condition occurs (as shown in Fig. 2). Hence, the goals of this research are: 1) to devise a new method that will help obtain a better tracking performance with laser scanner than those obtained from previous methods when the interactions occur; 2) to make a new attempt to solve the “merge/split” problem in the laser based multi-target tracking.

Mostly, multi-target tracking can be solved by data association [4]. Although a number of classical approaches to the problem of data association have been developed, most of them confront difficulties in dealing with complex interactions among targets. In this paper, we propose an on-line supervised learning based method for multi-target tracking. The essence of the proposed method is that learning and tracking can be integrated to deal with various complex tracking problems. The learning and tracking supplement each other in the proposed method: when the targets do not interact with each other, multiple independent trackers are utilized for the on-line learning; meanwhile, when the targets are in close proximity or a “merge/split” condition occurs, the learned classifiers are used to assist in tracking and deal with these challenging situations. Although some extra computation is needed for learning, the computational complexity of the proposed method increases linearly with the number of tracking targets. Hence, our method can be applied to tracking a large number of interacting targets, which is intractable for the methods using a joint state space representation.

Over the last couple of years, a large number of algorithms for multi-target tracking have been proposed. The multiple hypothesis tracker (MHT) [5] attempts to keep track of all the possible association hypotheses over time, which can be seen as the most successful algorithm based on data-oriented view. However, the computational complexity of MHT grows exponentially with the number of hypotheses. To tackle this problem, the nearest neighbor standard filter (NNSF) [4] associates each target with the closest measurement in the target space. However, this simple procedure prunes away many feasible hypotheses. In this respect, the joint probabilistic data association filter (JPDAF) [4,6], a suboptimal single-stage approximation to the optimal Bayesian filter, is more appealing. The main shortcoming of the JPDAF is that the final estimate is
Boosting is a popular supervised learning technique in the area of machine learning. The basic idea of boosting is to create an accurate and strong classifier by combining a set of weak classifiers. The requirement for each weak classifier is that its accuracy should be better than a random guessing. In this work, we choose to use the Gentle Adaboost algorithm introduced by Friedmen et al. [12] due to its numerically robustness. The input to the algorithm is a set of labeled training data \( (x_n, l_n), n = 1, ..., N \), where each \( x_n \) is an example and \( l_n \in \{+1, -1\} \) is the class label which indicates whether \( x_n \) is positive or negative respectively. Boosting learning provides a sequential procedure to fit additive models of form \( F(x) = \sum_{m=1}^{M} f_m(x) \). Here \( f_m(x) \) are called weak learners, and \( F(x) \) is a strong learner. The Gentle AdaBoost used adaptive Newton steps to minimize the cost function \( J = E[e^{wF(x)}] \), which corresponds to minimizing a weighted squared error at each step. We define the weak classifiers \( f_m(x) \) to be the Classification and Regression Trees [13]. More details about the Gentle Adaboost algorithm or boosting technique can be found in [12].

### B. Discriminative Features for Learning

In this subsection, we present six discriminative features for learning, which can be divided into motion features and walking pattern features. We believe that different persons have different walking habits. Moreover, different persons should have specific motion information (velocity, acceleration etc.) in a short term. This information can also be used for learning.

The training samples for the AdaBoost algorithm are given by a set of segments together with their labels

\[
E = \{(S, l) | l \in \{+1, -1\}\},
\]

where each segment \( S = \{(P_1, ..., P_n)_{foot1}, (P_{n+1}, ..., P_{2n})_{foot2}\} \) represents the two feet of one person, which contains some moving points \( P = (x, y) \) obtained by background subtraction. Details about how to cluster moving points to “two feet” can be found in our previous work [1,2].

We define a feature \( f \) as a function \( f : S \rightarrow \mathbb{R} \) that takes
a segment $S$ as an argument and returns a real value. For each segment $S$, we determine the following six features:

**Motion Features:**

1) Average velocity of segment: This feature measure the mean speed for a specific sample $i$, which is given by

$$v_i = \sqrt{(x_i^{(i+1)} - x_i^{(i)})^2 + (y_i^{(i+1)} - y_i^{(i)})^2} / \Delta t,$$  \hspace{1cm} (2)

where $x_i^{(i)}$, $y_i^{(i)}$ is the barycenter of segment $i$, $\Delta t$ the interval.

2) Velocity direction of segment: This feature measure the direction of velocity for a specific sample $i$, which is given by

$$\alpha_{v_i} = \arctan(v_{y,i} / v_{x,i}),$$  \hspace{1cm} (3)

where $v_{x,i}$ and $v_{y,i}$ are the average velocity on x-coordinate and y-coordinate.

3) Average acceleration of segment: This feature measure the mean acceleration for a specific sample $i$, which is given by

$$a_i = \sqrt{(v_{x,i}^{(i+1)} - v_{x,i}^{(i)})^2 + (v_{y,i}^{(i+1)} - v_{y,i}^{(i)})^2} / \Delta t.$$  \hspace{1cm} (4)

4) Acceleration direction of segment: This feature measure the direction of acceleration for a specific sample $i$, which is given by

$$\alpha_{a_i} = \arctan \left( \frac{v_{x,i}^{(i+1)} - v_{x,i}^{(i)}}{v_{y,i}^{(i+1)} - v_{y,i}^{(i)}} \right) / \Delta t.$$  \hspace{1cm} (5)

**Walking Pattern Features:**

When a normal pedestrians step forward, one of the typical appearances is, at any moment, one foot swings by pivoting on the other one. Two feet interchange their duty by landing and moving shifts at a rhythmic pattern. If we put all the laser points into the spatial-temporal domain, then it can be seen that some periodic braided styles are generated, as shown in Fig.1 (c). In [3], Shao et al. have proven that the walking period and walking stride of person $S$ becomes a parameter estimation problem:

$$S_j(t) = -\frac{1}{48} AT^2 \cos(2\pi t/T) = -S \cos(\gamma),$$  \hspace{1cm} (6)

where $A$ stands for the maximum acceleration, and $T$ for the walking period, $S$ for the stride, and $\gamma$ for the walking phase. Obviously, $S$ and $T$ can be used as discriminative features for learning. Although identifying left foot and right foot for a person is difficult, it is easy to compute $S_j$ for a segment. Hence, extracting the two features becomes very easy. For $n$ continued frames, we obtain the value of $S_j$ and $t_j$ (we assume they are in one periodic). Therefore, extracting features $S$ and $T$ becomes a parameter estimation problem:

$$S_j = \tilde{S} \cos(2\pi t_j/T), \hspace{1cm} j = 1,\ldots,n.$$  \hspace{1cm} (7)

We use a nonlinear least squares algorithm to estimate $\tilde{S}$ and $\tilde{T}$. More details about this algorithm can be found in [14]. On the other hand, in the tracking procedure, we also take $S$ and $T$ as state parameters to be estimated. Now, the walking pattern features can be determined as:

5) Walking stride of pedestrians $S$.

6) Walking period of pedestrians $T$.

Once we obtained a segment, the features mentioned above were extracted by several continued frames, which were used for learning or assisting in tracking.

**IV. BAYESIAN FORMULATION**

We formulate the sequential tracking problem as computing the maximum a posteriori (MAP) $x$ such that $x^* = \arg \max_{x,y} p(x | y), X = (x_0,\ldots,x_T)$ is the state sequence and $Y = (y_0,\ldots,y_T)$ the observation sequence. A basic Bayesian sequential estimation can be described as a two-step recursion:

**Prediction step:**

$$p(x_t | y_{t-1}) = \int p(x_t | x_{t-1})p(x_{t-1} | y_{t-1})dx_{t-1}. \hspace{1cm} (8)$$

**Filtering step:**

$$p(x_t | y_t) = \frac{p(y_t | x_t)p(x_t | y_{t-1})}{p(y_t | y_{t-1})}. \hspace{1cm} (9)$$

In this work, we employ the particle filter technique to compute for the Bayesian estimation. The basic idea behind the particle filter is very simple. Starting with a weighted set of samples $\{w_i, x_i\}_{i=1}^N$ approximately distributed according to $p(x_t | y_{t-1})$, new samples are generated from a suitably designed proposal distribution $q(x_t | x_{t-1})$, which is the dynamic model in this work. To maintain a consistent sample the new importance weights are set to

$$w_i^{(n)} \propto w_i^{(n-1)} q(x_t^{(n)} | x_{t-1}^{(n)}, y_t), \hspace{1cm} \sum w_i^{(n)} = 1. \hspace{1cm} (10)$$

More details about the particle filter can be found in [15].

Therefore, we provide details about state space of each target and dynamic model $p(x_t | x_{t-1})$ in the tracking below.

**State space and dynamic model:**

An individual person’s state at time $t$ is modeled using a four-dimensional state vector $x_{t,k} = [P_{t,k}, S_{t,k}, T_{t,k}]$, where $k$ is the person’s identifier, $P_{t,k}$ denotes the barycenter position of the “two feet segment” $S$, which is a two-dimensional vector $P_{t,k} = [x, y]$. $T_{t,k}$ and $S_{t,k}$ are the walking period and walking stride of person $k$ respectively. We do not use the position of human feet as state parameters as in our previous work [1,2], wherein we would need a complex dynamic model for prediction, which is not preferable to a strong observation.
Hence, we employ a constant velocity model in the tracking, which can be best described by a second order autoregressive equation

\[ P_{k,j} = AP_{k,j-1} + BP_{k,j-2} + CN(0, \Sigma). \]  

Matrices \( A \), \( B \), \( C \) and \( \Sigma \) could be obtained from a set of representative sequences, which are the constant ratio. \( N(0, \Sigma) \) is a Gaussian noise with zero mean and a standard deviation of 1. The prediction of \( T_{k,j} \), and \( S_{k,j} \) is performed by

\[ T_{k,j} = T_{k,j-1} + N_T, \]

\[ S_{k,j} = S_{k,j-1} + N_S, \]

where \( N_T \) and \( N_S \) are normal distributed random noise.

The details about observation model \( p(y_i | x_i) \) of targets will be provided in the next section as they are depending on different tracking situations.

V. ON-LINE LEARNING AND TRACKING

The learning and tracking supplement each other in our method. Before the update step, for each target, we should find its segment candidates and switch the tracking and learning from one to another. Therefore, the statistical distance function can be used:

\[ \frac{(x^*-x)^2}{G\sigma_x} + \frac{(y^*-y)^2}{G\sigma_y} = 1, \]  

where \( x^* \) and \( y^* \) is the predicted barycenter position of a person, \( G \) the threshold, \( \sigma_x \) and \( \sigma_y \) the covariance. In this section, we provide details about on-line learning and tracking.

A. Non-correlated Targets Tracking for Online Learning

If a target finds only one segment candidate and this segment is only possessed by this target, we assume this target does not interact with others. In this case, tracking this target becomes very easy. The observation is a point set in the predicted barycenter position, \( x^* \) and \( y^* \). The learned classifier above should be used: we use these segment candidates and the previous state of the target to extract features depicted in Section three, respectively. The features vector of segment candidates are inputted to the classifier of this target. The output of these classifiers is some scores. We normalize scores of different segment candidates into \([0,1]\) and ensure that they sum up to 1. These scores then become the probability used to measure the weight of this segment in the observation. Therefore, the observation for target \( k \) is:

\[ P(y_i | x_{kj}) = \sum_{j=1}^{M} \beta_{kj} P_j(y_{ij} | x_{kj}), \]  

where \( P_j(y_{ij} | x_{kj}) \) is the observation of candidate \( j \) computed by (17), \( \beta_{kj} \) the weights of candidate \( j \) depending on the classifiers, and \( M \) is the number of segment candidates. Lastly, the state of this target is obtained after particle filtering process.

Sometimes, it is difficult to associate two legs to one person, for instance, extracting good features for labeling is very difficult when there are three legs. This condition occurs, we deal with it as a merge condition depicted below.

Merge/Split condition:

If a segment candidate is shared by more than one target or if we do not have enough confidence to associate two legs to one person (several people stand close together), then we assume that different targets merge together. In this case, we track this merged segment as one target which is similar to

After re-weighting and re-sampling in the particle filter, we save the state of this target. Then we extract features depicted in the section three in some continued frames and put them into the sample pool for training a classifier.

Pedestrians sometimes change their walking styles. Consequently, using an unchangeable classifier for a long term is not a wise solution. Therefore, we should update the classifier periodically. Fig. 6 shows the life-spans of a classifier: once we trained a classifier, it was put in service for a specific time, and a sample pool was collected samples for the next training synchronously. In this case, as the classifier was updated periodically, we can ensure the validity of the classifier to distinguish different persons.

B. Learning for Tracking Correlated Targets

Once challenging conditions occur, the learned classifiers should assist in tracking.

Interacting condition:

If a target finds more than one segment candidates, we assume this target interacts with other persons. In this case, independent tracker cannot provide reliable results any longer. Hence, the learned classifiers above should be used: we use these segment candidates and the previous state of the target to extract features depicted in Section three, respectively. The features vector of segment candidates are inputted to the classifier of this target. The output of these classifiers is some scores. We normalize scores of different segment candidates into \([0,1]\) and ensure that they sum up to 1. These scores then become the probability used to measure the weight of this segment in the observation. Therefore, the observation for target \( k \) is:

\[ P(y_i | x_{kj}) = \sum_{j=1}^{M} \beta_{kj} P_j(y_{ij} | x_{kj}), \]  

where \( P_j(y_{ij} | x_{kj}) \) is the observation of candidate \( j \) computed by (17), \( \beta_{kj} \) the weights of candidate \( j \) depending on the classifiers, and \( M \) is the number of segment candidates. Lastly, the state of this target is obtained after particle filtering process.

By this condition, we track this merged segment as one target which is similar to
non-correlated target tracking. Once the merged segment finds more than one segment candidates, we believe that this merged segment splits to different sub-segments, we track these sub-segments respectively as non-correlated target tracking. Then we extract features and use the classifiers to identify them. Lastly, we link these trajectories to one for each target.

C. Target Addition or Removal

Our method is also suitable for tracking a variable number of targets. The target addition and removal is depending on the statistical function. If a target on the edge of a coordinate plane cannot find any segment candidate for some continued frames, we assume this target may disappear. We save the trajectory of this target, and stop to track this target. Similarly, if a “moving segment” on the edge of a coordinate plane cannot find any target for some continued frames, then this should be a new target. A new sample pool for learning is assigned to this target and the particles for this target are initialized.

VI. EXPERIMENTS AND RESULTS

We evaluated the proposed method in the real scene at the plaza (about 30m × 25m), crossing, and corridor. One single-row laser scanner, LMS291, produced by SICK was utilized. The laser scanner was set on ground surface performing horizontal scanning with a frequency of 37 fps. The selected data used for testing were seven different clips consisting of more than 12000 frames in which complex interactions frequently took place. The results and some comparisons are detailed in this section.

A. Tracking Performance

In this subsection, we present our tracking results and make some analysis about the tracking process.

Fig. 7 and Fig. 8 give a vivid view of our tracking process. In the frame 560, target 4 and target 5 were in close proximity. In this case, the observation of each target contained two segments, the weights of which were dependent on its classifier. For each target, the position observation was more peaked around its real position. As a result the particles were more focused around the true target state after each level’s re-weighting and re-sampling. After 20 frames, we only obtained one segment for two targets, in which we believed that the two targets were merging. In this case, we tracked this segment as one target and no longer used the classifier. Hence, the components in this observation had the same weights. After 30 frames the two targets splitted. We maintain their correct identifications using the classifiers. The final trajectories of these targets were shown in Fig. 9.

Fig. 10 shows the tracking performance under a more challenging situation. In frame 95, three persons were walking together. In frame 183, they were merging, thus we tracked the three persons as a one target. After 9 frames, one target splitted out. We made a correct labeling with the help of the classifier and continued tracking targets 2 and 3 as one target. In frame 213, targets 2 and 3 splitted, and the classifiers of these targets were used again. At last, we maintained the correct trajectories of these targets with the proposed method.

Table 1 shows the quantitative performance of our method for tracking seven different clips. In these clips, our method failed occasionally due to three conditions: (1) Some new appearing targets were the interacting targets, which made our method be not initialized correctly and made the learning become difficult. (2) Some targets were occluded by others for a long time. (3) The motion of some merging targets changed greatly when they split.

B. Quantitative Comparison

A quantitative comparison was conducted among four methods: Joint Probabilistic Data Association Filter (JPDAF), Monte Carlo Joint Probabilistic Data Association Filter (MC-JPDAF), Nearest Neighbor Standard Filter (NNSF) and our method.

We made a statistical survey of 1000 continued frames to evaluate the tracking performance of these methods in the condition of interactions. We defined that if the distance between two persons was less than 30cm, one interaction would be counted. Because JPDAF can only track a fixed number of targets, we started tracking when target numbers were more than five and re-initialized all the algorithms when the target numbers changed. We used 500 particles per filter in the simulation and the ground truth was obtained by manual labeling. Fig. 11 shows the quantitative evaluation of the four methods. We can see that our method has a better performance than the other three methods. In frame 776, the
trajectories of eight persons obtained by our method were quite approximate to the ground truth. Compared to our method, we found that it was difficult for the other methods to rectify its wrong tracking once they made an incorrect data association. From frame 723 to 783, JPDAF continued to its wrong tracking of targets 1 and 3. The success rates of the four methods are shown in Table 2.

VII. CONCLUSION

In this paper, a novel on-line supervised learning based method is presented for tracking interacting targets. Different evaluations show the superior tracking performance of the proposed method under some complex situations. Of course, our method also has some limitations: (1) Our method is not suitable to some small and cabined environment. (2) Our method should face continued failed tracking once the classifiers make a wrong labeling. This is a preliminary work to introduce the existing learning technique into the laser based multi-target tracking problem. In the future, more effective strategies should be explored to deal with these problems.

ACKNOWLEDGMENT

This work was supported in part by the NKBRPC (No. 2006CB303100), NSFC Grant (No. 60333010), NSFC Grant (No. 60605001), the NHTRD P 863 Grant (No. 2006AA01Z302), Key Grant Project of Chinese Ministry of Education (No. 103001) and NHTRDP 863 Grant (2007AA11Z225).

REFERENCES