An Online System for Multiple Interacting Targets Tracking: Fusion of Laser and Vision, Tracking and Learning

XUAN SONG, Key Laboratory of Machine Perception (MoE), Peking University; The University of Tokyo
HUJING ZHAO, Key Laboratory of Machine Perception (MoE), Peking University
JINSHI CUI, Key Laboratory of Machine Perception (MoE), Peking University
XIAOWEI SHAO, The University of Tokyo
RYOSUKE SHIBASAKI, The University of Tokyo
HONGBIN ZHA, Key Laboratory of Machine Perception (MoE), Peking University

Multi-target tracking becomes significantly more challenging when the targets are in close proximity or frequently interact with each other. This paper presents a promising online system to deal with these problems. The novelty of this system is that laser and vision are integrated with tracking and online learning to complement each other in one framework: when the targets do not interact with each other, the laser-based independent trackers are employed and the visual information is extracted simultaneously to train some classifiers online for “possible interacting targets”. When the targets are in close proximity, the classifiers learned online are used alongside visual information to assist in tracking. Therefore, this mode of co-operation not only deals with various tough problems encountered in tracking, but also ensures that the entire process can be completely online and automatic. Experimental results demonstrate that laser and vision fully display their respective advantages in our system, and it is easy for us to obtain a good trade-off between tracking accuracy and the time-cost factor.

Categories and Subject Descriptors: I.4.9 [Image Processing and Computer Vision]: Applications
General Terms: Measurement, Algorithms

ACM Reference Format:
DOI = 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

1. INTRODUCTION

A robust and efficient multi-target tracking system has become an urgent need in various application domains, such as surveillance, intelligent transportation, and pedestrian flow analysis. Compared to a traditional camera-based tracking system, as a new kind of measurement instrument, the laser range scanner (LIDAR) has received increasing attention for solving tracking problems in recent years. In a laser-based tracking system (as shown in Fig. 1), the targets are represented by several points; hence, the tracking becomes much easier, and it is simple to obtain significantly better performance in terms of both accuracy and the time-cost factor when the targets are far apart. The system [Zhao and R.Shibasaki 2005; Song et al. 2010] has been successfully applied to the JR
Fig. 1. A typical laser-based tracking system.

Fig. 2. How can the persons’ correct trajectories be maintained under this condition? In frame 105, three people are walking together. Their trajectories merge in frame 175; how can their correct trajectories be maintained when they split?

subway station of Tokyo for the analysis of pedestrian flow, and obtained an overall accuracy of 83%.

However, the drawback of a laser-based tracking system is inherent and obvious: it lacks visual information, and it is consequently difficult to obtain a set of features that uniquely distinguish one object from another. Hence, when the targets are in close proximity or frequently interact with each other, robust tracking becomes especially challenging. Moreover, when the well-known “merge/split” situation occurs (as shown in Fig. 2), maintaining the correct tracking seems to be impossible. It is easy to consider the combining of lasers and vision into one framework to solve these problems. Therefore, the core concerns of this research are:

1. How to make the laser and vision fully display their respective advantages in one framework to solve the difficult problems encountered in multi-target tracking?
2. How to develop a tracking system that can obtain a good trade-off between tracking accuracy and the time-cost factor?

In this paper, we integrate lasers, vision, tracking, and online learning such that they complement each other in one framework to deal with various tracking problems. The key idea of this work is depicted in Figs. 3 and 4. When the targets do not interact with each other, the laser scanner can efficiently track them, and it is easy to extract visual information from the camera data. Due to the reliability of these tracking results, they are used as positive or negative samples to train some classifiers for the situation of “possible interacting targets”. When the targets are in close proximity, the learned classifiers and visual information will in turn assist in tracking.

This mode of co-operation between lasers, vision, tracking, and learning offers several advantages: (1) Lasers and vision can fully display their respective advantages (the speed of a laser and
Learning by Tracking. When the targets do not interact with each other, the laser-based independent trackers are employed and the visual information is extracted simultaneously to perform the on-line learning.

Tracking by Learning. When the targets are in close proximity, the learned classifiers and visual information in turn assist in tracking.

the rich information of a camera) in this system. (2) Because the “possible interacting targets” are depicted by a discriminative model with an online supervised learning process, this model can consider information from the “confusing targets” and sufficiently exploit the targets’ history. Through these discriminative models, we can easily deal with some challenging tracking situations. (3) This “tracking-learning adaptive loop” ensures that the entire process can be completely online and automatic.

2. RELATED WORK

Multiple target tracking (MTT) has been studied extensively, and an in-depth review of tracking literature can be found in a recent survey by Yilmaz et al. [Yilmaz et al. 2006]. Typically, multi-target tracking can be solved through data association [Bar-Shalom and Fortmann 1998; Bar-Shalom and Li 1995]. The multi-hypothesis tracker (MHT) [Read 1979; Blackman and Popoli 1999; Arras et al. 2008], which attempts to keep track of all possible association hypotheses over time, can be seen as the most successful algorithm from a data-oriented view. But the MHT algorithm is computationally exponential, both in memory and time, which makes it difficult to apply. The nearest neighbor standard filter (NNSF) [Bar-Shalom and Fortmann 1998] associates each target with the closest measurement in the target state space, and employs the particle filter [Doucet et al. 2000] or Kalman filter [Broida and R.Chellappa 1986; Fod et al. 2002] to complete its tracking. However, this simple procedure prunes away many feasible hypotheses and cannot solve “labeling problems” when the targets are crowded. In this respect, a wider approach to multi-target tracking is achieved.

ACM Transactions on Intelligent Systems and Technology, Vol. V, No. N, Article A, Publication date: January YYYY.
by exploiting a joint state space representation that concatenates all of the targets’ states together [Okuma et al. 2004; Vermaak et al. 2003; Zhao and Nevatia 2004] or infers this joint data association problem by characterizing all of the possible associations between the targets and observations. Examples of this approach include the joint probabilistic data association filter (JPDAF) [Bar-Shalom and Fortmann 1998; Rasmussen and Hager 2001; Gemmari and Hager 2004], Monte Carlo technique based JPDA algorithms (MC-JPDAF) [Vermaak et al. 2005; Schulz et al. 2003], and Markov chain Monte Carlo data association (MCMC-DA) [Oh et al. 2004; Khan et al. 2006; Yu et al. 2007]. In addition, researchers have also proposed to utilize Random Finite Set (RFS) approach [Goodman et al. 1997; Mahler 2007b] to deal with multi-target tracking and sensor fusion problem. This methodology offers an elegant Bayesian formulation of multi-target filtering, which has resulted in the development of the Probability Hypothesis Density (PHD) filter [Mahler 2003] and the Cardinalized PHD (CPHD) filter [Mahler 2007a]. More recently, some efficient implementations of these filters have been proposed and received the increasing attentions, such as Gaussian mixture probability hypothesis density filter (GM-PHD) [Vo and Ma 2006], Gaussian mixture Cardinalized probability hypothesis density filter (GM-CPHD) [Ulmeke et al. 2010], Multi-Bernoulli filter [Vo and Ma 2009] and sequential Monte Carlo (SMC) methods [Vo et al. 2005]. Moreover, researchers also propose some global optimization strategies [Jiang et al. 2007; Leibe et al. 2008; Zhang et al. 2008; Shitrit et al. 2011] to reduce complexity and optimize data association algorithms. Recently, the “low-frequency” terms of a Fourier decomposition were used to represent distributions over data association [Huang et al. 2009]. This method can maintain and update the permutation distribution directly in the Fourier domain, allowing for polynomial time band-limited approximations.

Additionally, researchers have proposed the use of multiple parallel filters to track multiple targets [Cai et al. 2006; Shao et al. 2007]; that is, one filter per target where each has its own small state space. In spite of this, when interactions occur among targets, this method encounters difficulty in maintaining the correct tracking. Therefore, modeling the interactions among targets becomes an incredibly important issue. Khan et al. [Khan et al. 2005] used a Markov random field (MRF) motion prior to modeling the interactions among targets. Qu et al. [Qu et al. 2005] proposed a magnetic-inertia potential model to handle the “merge error” problem. Yu et al. [Yu and Wu 2005] allowed for collaboration among filters by modeling the targets’ joint motion prior to using a Markov random network to solve the “merge error” problem. Lanz et al. [Lanz and Manduchi 2005] proposed a hybrid joint-separable model to deal with the interactions among targets. Sullivan et al. [J.Sullivan and S.Carlsson 2006] tracked isolated and “merging targets” separately, and then connected their trajectories by a clustering procedure. Nillius et al. [P.Nillius et al. 2006] employed a track graph to describe when targets are isolated and how they interact. They utilized a Bayesian network to associate the identities of the isolated tracks by exploiting the graph.

Meanwhile, researchers found that it was somewhat difficult to obtain robust tracking results with a single sensor. Therefore, several tracking systems based on the combination of a laser and a camera have been proposed [Arras et al. 2000; Chakravarty and Jarvis 2006; Scheutz et al. 2004; Blanco et al. 2003; Schulz 2006; Bellotto and Hu 2006; Cielniak et al. 2007]. The system of of Arras et al. [Arras et al. 2000] integrates 360 degree laser and monocular vision to perform localization in a mobile robot, whereas that of Chakravarty and Jarvis [Chakravarty and Jarvis 2006] utilized the panoramic vision and a laser finder to track people in an indoor environment. Scheutz et al. [Scheutz et al. 2004] utilized a combination of leg detection, based on distance information from a laser range sensor, and a visual face detector to track people using a mobile robot. However, all of the above systems can only detect and track one or a few persons, which is not sufficient in a real surveillance and monitoring environment. The systems of Cielniak et al. [Cielniak et al. 2007] and Blanco et al. [Blanco et al. 2003] used classifiers to assist in tracking, but their classifiers are pre-trained detectors that cannot be updated during the tracking, i.e., a completely off-line system of learning. In addition, the work described in [Oliveira et al. 2010; Spinello and Siegwart 2008; Song et al. 2008; Cui et al. 2008] can be utilized to detect or track more targets in the outdoor environment, but once some challenging situations (such as “merge/split”) occur, these systems have difficulty maintaining the correct tracking.
More recently, there has been a trend toward introducing online learning techniques into target tracking. Representative publications include [Avidan 2007; Li et al. 2007; Lu and Hager 2006; Grabner and Bischof 2006; Kuo et al. 2010; Song et al. 2008b]. [Avidan 2007; Grabner and Bischof 2006] formulated the tracking task as a classification problem, and continuously update the current boosting classifier that represents the object in order to optimally discriminate it from the current background. Lu et al. [Lu and Hager 2006] utilized random image patches to represent the foreground of an image, and then extract these from the background via an online updating process. Li et al. [Li et al. 2007] proposed a particle filter-based tracker with different online learning observations. [Song et al. 2008b; Kuo et al. 2010] utilize some online trained classifiers to describe the appearance model of targets, and try to deal with interaction and occlusion among targets.

Therefore, in this paper, we propose a novel, online tracking system that integrates these promising technologies. Our system has the following key features that make it advantageous over previous ones: (1) It can be applied in wide and open areas, and can easily deal with interactions and merge/split problems among targets. (2) It enables laser and vision to fully display their respective advantages in the overall system, is faster than traditional vision-based tracking systems, and more robust than traditional laser-based ones. (3) The proposed system performs in a fully online and automatic way. It can incrementally learn and update targets’ appearance model, and there is no need for manual labeling.

3. PROBLEM FORMULATION AND SYSTEM OVERVIEW

We begin by introducing notations to formulize our problem. At time $t$, pedestrian $k$ is represented by $x_{k,t} = [p_{L,k,t}, p_{R,k,t}, T_{k,t}, S_{k,t}, a_{L,k,t}, a_{R,k,t}]$, where $p_{L,k,t}$ and $p_{R,k,t}$ denote the positions of the person’s two feet, which is a two-dimensional vector $p_{k,t} = (x, y)$. $T_{k,t}$ and $S_{k,t}$ are the walking period and the walking stride of a person, respectively. $a_{L,k,t}$ and $a_{R,k,t}$ are the accelerations of their feet. Based on the laser or visual measurements $z_t$, we need to estimate its state as

$$\hat{x}_{k,t} = \arg \max_{x_{k,t}} p(x_{k,t} | z_t).$$

(1)

The posterior probability $p(x_{k,t} | z_t)$ can be computed by a Bayesian recursion as

$$p(x_{k,t} | z_t) = \int p(z_t | x_{k,t}) p(x_{k,t} | x_{k,t-1}) p(x_{k,t-1} | z_{t-1}) dx_{t-1},$$

(2)

where $\gamma$ is the normalization constant, $p(z_t | x_{k,t})$ is the similarity between the target’s state and measurements (observation model), and $p(x_{k,t} | x_{k,t-1})$ is the transition probability (dynamic model). In this research, we utilize the “two feet walking” model [Shao et al. 2007] as a dynamic model to predict pedestrians’ position, and employ the particle filter technique [Doucet et al. 2000] to compute Equation (1) and (2). When the pedestrians are in the far distance (non-correlated targets), the observation model is mainly based on the laser measurements. When the pedestrians are in close proximity or interact with each other (interacting targets), the observation model is based on the classifier-enhanced visual observations.
Therefore, the overall tracking system is designed as illustrated in Fig. 5 and Fig. 6, and mainly consists of four components: a sensor fusion module, a tracking situation switch module, a non-correlated target tracking module (learning by tracking), and an interacting target tracking module (tracking by learning). In the sensor fusion part (as shown in Fig. 5-a), a number of laser scanners and one video camera are exploited. Each of these is controlled using a client computer. All client computers are connected through a network to a server computer, which synchronizes and integrates the data from all client computers. For data synchronization, each laser scan and video stream is stamped with a time log at the moment it is captured, or starts to be captured, using the client computer’s local clock, which is corrected periodically according to that of the server computer. Data measured by different client computers that is stamped with the same time log is aligned to make up an integrated frame. In addition, an extrinsic calibration is conducted by several control points in a box (as shown in Fig. 5-b). Based on the extrinsic calibration, we can easily compute the center position of the bounding box in image coordinates through the “two feet” position. For more details about this part, please refer to our previous work [Cui et al. 2008].

The main components that will be described in this paper are the tracking situation switch, the non-correlated target tracking (learning by tracking), and the interacting target tracking (tracking by learning). In the following sections, we provide details about how the lasers, vision, tracking, and learning complement each other in one framework.

4. TRACKING SITUATION SWITCH
We examine different situations that are encountered in tracking, such as non-correlated target tracking, interacting target (either in close proximity or merging with each other) tracking, splitting of “merged targets”, and the appearance or disappearance of targets. We aim to detect these conditions and make the system switch automatically between tracking and learning. In this research, we employ a detection-driven strategy to perform this task.

In each frame, we must obtain a laser-based detection of the targets to aid in determining the tracking situation. Each detection or target is modeled as \( \text{Seg} = \{ (P_1, \ldots, P_n)_{\text{foot} 1}, (P_1, \ldots, P_n)_{\text{foot} 2} \} \), which represents the two feet of one person. The \( P = (x, y) \) are the moving points that are obtained by background subtraction. We use the Mean-shift algorithm, as described in [Comaniciu and Meer 1999], to cluster these moving points into a “two feet segment” (as shown in Fig. 7-a).

We define four conditions for each target: a non-correlated target condition, a correlated target condition, a merge/split condition, and an appearance/disappearance of targets condition. The correlated targets and merge/split targets can be seen as interacting targets. A statistical distance function...
where $\sigma$ is defined as:

$$\sigma^2 = \frac{(x'_{k,t} - x)^2}{G^2\sigma_x^2} + \frac{(y'_{k,t} - y)^2}{G^2\sigma_y^2} = 1$$

where $d_{k,t}^e = (x'_{k,t}, y'_{k,t})$ is the predicted barycenter position of target $k$ in frame $t$, $G$ is the threshold, and $\sigma_x$ and $\sigma_y$ are the covariance. We set the threshold $G = 2\rho$, where $\rho$ is the bandwidth of Mean-shift clustering. Hence, we can utilize this function to search for the possible detections of targets: for a “two feet segment” $Seg$ with its cluster center $(x_{seg}, y_{seg})$, if

$$\frac{(x'_{k,t} - x_{seg})^2}{G^2\sigma_x^2} + \frac{(y'_{k,t} - y_{seg})^2}{G^2\sigma_y^2} \leq 1,$$

this segment will associate to this target, and vice-versa.

Meanwhile, for each target, we assign a variable $c \in \{0, 1, 2\}$ to indicate its association state, which is defined as:

$$c = \begin{cases} 
0, & \text{if associated to no detection} \\
1, & \text{if associated to one detection} \\
2, & \text{if associated to more than one detection}
\end{cases}$$
Similarly, for each detection, we assign an association state $r \in \{0, 1, 2\}$ according to:

$$
\begin{align*}
  r &= \begin{cases} 
    0, & \text{if associated to no target} \\
    1, & \text{if associated to one target} \\
    2, & \text{if associated to more than one target}
  \end{cases}
\end{align*}
$$

An example is shown in Fig. 7-b,c. In Fig. 7-b, the target utilizes the Equation (3) to find two detections, hence its association state $c = 2$. Meanwhile, the two detections are only associated with this target, and so their association state $r = 1$. In contrast, in Fig. 7-c, the target only finds one detection ($c = 1$), and the association state of this detection is $r = 1$.

Therefore, these tracking situations can be categorized into four different states:

**Non-correlated targets:** Given target $k$ in frame $t$, if $c_{k,t} = 1$ and its associated detection has $r_{j,t} = 1$, then this is a non-correlated target (as shown in Fig. 7-c).

**Correlated targets:** Given target $k$ in frame $t$, if $c_{k,t} = 2$ and it is not a “merged target”, we believe that there is more than one detection having an influence on this target. Hence, we treat it as a correlated target (as shown in Fig. 7-b).

**Merge/Split condition:** Given detection $j$ in frame $t$, if $r_{j,t} = 2$ and all of its associated targets have $c_{k,t} = 1$, we believe that these targets are merging. Hence, we initialize a new state space for this “merged target” (as shown in Fig. 8-b). Moreover, if this merged target has $c_{k,t} = 2$ and all of its associated detections have $r_{k,t} = 1$, we believe that this “merged target” should split (as shown in Fig. 8-c). Note that when a non-merged target splits, we do not consider this to be a merge/split condition (e.g., where several people appear in the frame together before later splitting).

**Appearance/Disappearance of targets:** Given a target that is on the edge of the coordinate plane, if $c_{k,t} = 0$ we assume that this target should disappear. We save the state of this target and stop tracking it. Similarly, for a detection that is on the edge of the coordinate plane, if $r_{k,t} = 0$ this becomes an appearing target. Hence, we should assign a new state space for this target and start to track it.

Although this is a simple strategy, it is effective for detecting most of the conditions described above, and can automatically switch between tracking and learning.

5. LEARNING BY TRACKING

Actually, when the targets are far apart (non-correlated), tracking becomes relatively easy as it can be solved through multiple laser-based independent trackers. Moreover, laser scanners are insensitive to weather conditions, and the data is easy to process; therefore, this situation can provide robust and efficient tracking. Once we obtain the tracking results, visual information for this target can be extracted from the camera and used as samples to train a classifier. The positive samples should be from this target, and the negative samples should come from the “possible confusing targets”. Consequently, the classifier of each target is a discriminative model that not only depicts the appearance of the target, but also considers information from other targets. An example is shown in Fig. 9. In this section, we provide further details about this learning by tracking.

5.1. Laser-based Independent Trackers

Multiple independent particle filter-based trackers are employed to perform laser-based tracking for non-correlated targets. For each target, we utilize the independent particle filter to compute Equations (1) and (2), and the data association is through the Nearest Neighbor (NN) association. A “two feet walking model” [Shao et al. 2007] is utilized as the proposal distribution. The observation is a point set in this segment expressed as $z_t = \{(x_i^t, y_i^t) : i = 1, ..., N\}$, and the observation model is the similarity between laser point sets and the predicted walking style of pedestrians.
Fig. 9. Non-correlated targets tracking and the on-line supervised learning. (a) We employ the laser-based independent trackers to track targets A and B and detect that they should be “possible interacting targets” in frame 8954. (b) Some random image patches are sampled in them, and the visual information is extracted. (c) These image patches are used as positive or negative samples to train the classifiers A and B.

For target \(k\), the similarity between the positions of the human feet and their nearest laser points cluster \(j\) can be described as

\[
P^{L}_{pos}(z_{t}|x_{k,t}) = \frac{1}{\sqrt{2\pi}\delta_{p}} \times \exp \left( -\frac{\left( x^{*}_{L} - x^{j}_{L,t}\right)^{2} + \left( y^{*}_{L} - y^{j}_{L,t}\right)^{2}}{2\delta_{p}^{2}} \right),
\]

\[
P^{R}_{pos}(z_{t}|x_{k,t}) = \frac{1}{\sqrt{2\pi}\delta_{p}} \times \exp \left( -\frac{\left( x^{*}_{R} - x^{j}_{R,t}\right)^{2} + \left( y^{*}_{R} - y^{j}_{R,t}\right)^{2}}{2\delta_{p}^{2}} \right),
\]

where \(P^{*} = (x^{*}_{L}, y^{*}_{L})\) and \(P^{*} = (x^{*}_{R}, y^{*}_{R})\) are the predicted position of human feet. To find the nearest laser points cluster, we use Euclidean distance here to measure the distance between the barycenter of each cluster and the predicted position of the feet.

On the other hand, Shao et al. [Shao et al. 2007] have proved that the distance between two feet varies in a cosine pattern which is simply the cosine function:

\[
S_{d}(t) = -\frac{1}{48} |A|T^{2}\cos\left(\frac{T}{T_{k}}\right) = -S \cos(\gamma),
\]

where \(A\) stands for the maximum acceleration, and \(T\) is the walking period, \(S\) is the stride, and \(\gamma\) is the walking phase. Hence, the walking pattern similarity can be described as:

\[
P_{walk}(z_{t}|x_{k,t}) = \frac{1}{\sqrt{2\pi}\delta_{w}} \times \exp \left( -\frac{\left( S_{d} - S_{k,t}\cos(2\pi T_{k}/T_{k,t})\right)^{2}}{2\delta_{w}^{2}} \right),
\]

where \(S_{d}\) is the distance between the “two-feet” position, which can be automatically computed by laser measurements.

We assume that these similarities are independent of each other in a short interval, an assumption that has been demonstrated to be workable in our previous research [Shao et al. 2007; Cui et al.
Hence, the likelihood function to be updated in the filter is:

\[
P_{\text{laser}}(z_t|x_{k,t}) = P_{\text{pos}}^L(z_t|x_{k,t}) \\
\times P_{\text{pos}}^R(z_t|x_{k,t}) \times P_{\text{walk}}(z_t|x_{k,t}).
\]  

(10)

Hence, after the re-weighting and re-sampling in the particle filter process, we obtain the new position of each pedestrian.

5.2. Visual Information Extraction

Once we obtain the tracking results for each target (the position of their two feet), they can be projected onto the image coordinates, and the bounding box (position and scale) of the people being tracked can be automatically computed through the calibrated parameters (for details about extrinsic calibration, please refer to [Cui et al. 2008]). A set of random image patches [Lu and Hager 2006] are then spatially sampled within the bounding box of the target. These random image patches all have the same size, and the sampling number is dependent on the scale of the bounding box. We utilize these random image patches as samples for the online supervised learning. In this case, each target is represented by a “bag of patches” model.

Extracting some distinguishable features from the image patches is relatively important for the learning process. There are a large number of derived features that can be employed to represent the appearance of an image patch, such as color information (RGB intensity vector), texture information (Haralick features), and edge information (edge orientation histogram). As we utilize image patches as samples, the feature vector should contain local information. Hence, we employ the RGB intensity + edge orientation histogram (EOH) [Levi and Weiss 2004] to extract features from image patches (as shown in Fig. 9-b). We adapt a d-dimensional feature vector to represent each image patch. Therefore, these feature vectors can be utilized as samples for learning or testing.

5.3. On-line Learning and Updating

The “possible interacting targets” each require strong classifiers to be trained, representing the appearance of these targets. In this research, we choose to use “Boosting” as the learning technique, and employ “Classification and Regression Trees” [Breiman et al. 1984] as weak classifiers. The basic idea of Boosting is to create an accurate and strong classifier by combining a set of weak classifiers. This is a popular supervised learning method in the area of machine learning. Let each image patch be represented as a d-dimensional feature vector. For target \( k \) in frame \( t \), \( \{s_{k,t}, l_{k,t}\}_{i=1}^N \) denote \( N \) samples and their labels, where \( s \in \mathbb{R}^d \) and \( l \in \{-1, +1\} \). The positive samples are the image patches from the bounding box of target \( k \), whereas the negative samples are the image patches from some “possible confusing targets”. In this work, the \( L \) closest targets to person \( k \) are seen as the “possible confusing ones”. Because the learning is performed while the targets are in the far distance and are not interacting with each other, the laser-based trackers are relatively robust and can easily avoid the “drift problem”. Moreover, the system can also ensure that the training patches do not become contaminated with the incorrectly labeled examples. An example is shown in Fig. 9: in frame 8954, persons A and B are walking towards each other, and should become “possible interacting targets”. Hence, classifiers A and B are trained for them. Obviously, for classifier A, the positive training image patches are from person A, and the negative ones are from person B, and vice-versa for classifier B. We can see that the training image patches can be correctly labeled by the system itself without any human intervention.

As time passes, new image patches from targets become available. In order to make the classifier increasingly stronger and reflect changes in the object’s appearance, the strong classifier should be updated synchronously. Therefore, we employ a strategy, motivated by [Avidan 2007], to deal with these problems: once the new training samples are available, we keep the \( K \) best weak classifiers, and add some new weak classifiers to replace the “not-working” ones. In this case, the learned classifiers not only reflect variations in the targets’ appearance model, but also maintain historical
Learning Algorithm

**Input:** Feature vectors of targets and their labels \( \{s_{k,t}^i, l_{k,t}^i\}_{i=1}^N \), for \( t = 1, \ldots, T \).

**Output:** The strong classifier \( H(s_{k,t}) \) of target \( k \) at time \( t \).

**Train a Strong Classifier** (for time 1):
1) Initialize weights \( \{w_i\}_{i=1}^N \) to be \( 1/N \).
2) For \( j = 1 \ldots M \) (train \( M \) weak classifiers)
   a) Make \( \{w_i\}_{i=1}^N \) a distribution.
   b) Train a weak classifier \( h_j \).
   c) Set \( \text{err} = \sum_{i=1}^N w_i |h_j(s_{k,1}^i) - l_{k,1}^i| \).
   d) Set weak classifier weight \( \alpha_j = 0.5 \log(1 - \text{err})/\text{err} \).
   e) Update example weights \( w_i = w_i e^{\alpha_j |h_j(s_{k,1}^i) - l_{k,1}^i|} \).
3) The strong classifier is given by \( H(s_{k,1}) \), where \( H(s_{k,1}) = \sum_{j=1}^M \alpha_j h_j(s_{k,1}) \).

**Update the Strong Classifier** (for time \( t \))
1) Initialize weights \( \{w_i\}_{i=1}^N \) to be \( 1/N \).
2) For \( j = 1 \ldots K \) (choose \( K \) best weak classifiers and update their weights)
   a) Make \( \{w_i\}_{i=1}^N \) a distribution.
   b) Choose \( h_j(s_{k,t-1}) \) with minimal \( \text{err} \) from \( \{h_1(s_{k,t-1}), \ldots, h_M(s_{k,t-1})\} \).
   c) Update \( \alpha_j \) and \( \{w_i\}_{i=1}^N \).
   d) Remove \( h_j(s_{k,t-1}) \) from \( \{h_1(s_{k,t-1}), \ldots, h_M(s_{k,t-1})\} \).
3) For \( j = K + 1 \ldots M \) (add new weak classifiers)
   a) Make \( \{w_i\}_{i=1}^N \) a distribution.
   b) Train a weak classifier \( h_j \).
   c) Compute \( \text{err} \) and \( \alpha_j \).
   d) Update example weights \( \{w_i\}_{i=1}^N \).
4) The updated strong classifiers are given by \( H(s_{k,t}) \), where \( H(s_{k,t}) = \sum_{j=1}^M \alpha_j h_j(s_{k,t}) \).

---

**Fig. 10.** On-line learning algorithm

information about targets and become increasingly stronger. The overall learning algorithm is shown in Fig. 10.

6. TRACKING BY LEARNING

When the targets are in close proximity or interact with each other, it is difficult for the laser-based independent tracker to maintain correct tracking. Specifically, when the “merge/split” condition occurs, associating the identities of the targets becomes a somewhat challenging problem. In this case, visual information and the learned classifiers should help the system to deal with these problems. In this section, we provide details about how these classifiers assist in tracking to deal with such challenging situations.

6.1. Correlated Targets Tracking

As explained in Section 4, once a target has more than one detections, and is not a “merged target”, we conclude that there are some correlated targets. When this condition occurs, a set of random image patches are sampled within the interacting region of the image, and the feature vectors of these
in image coordinates, which are dependent on the extrinsic calibration parameters. Hence, we convert the “two feet position” and the persons’ state into the center position of the bounding box, which is the center position of candidate target $i$. Therefore, we can easily obtain the new position of these persons after each level’s re-weighting and re-sampling. Finally, we utilize the weighted average values of these particles as the estimated position of the person, and this was usually close to the real center position of the person’s bounding box. Subsequently, we can easily obtain the new position of these interacting targets and avoid the “drift problem”. An overview of the process is shown in Fig. 11.

6.2. Merge/Split Condition

Sometimes, it is difficult to obtain separate detections with the laser-based clustering algorithm. Moreover, the targets in an image may occlude each other completely. Hence, the solution described above is not available. Once we detect that such a situation has occurred (as discussed in Section 4), we deal with it as a “merge/split” condition.

If some targets are merging together, we initialize the state of the “merging targets” and track them as one target. If we detect that this “merging target” has split and become an interacting or correlated target, we can track them as separate targets. However, if we detect that some targets are merging together, we initialize the state of the “merging targets” and track them as one target. If we detect that this “merging target” has split and become an interacting or correlated target, we can track them as separate targets.

**Correlated targets tracking.** (a) We detect that targets A and B are correlated targets. (b) Some random image patches are sampled in them. (c) We use their classifiers to obtain the score maps. (d) After the particle filtering process, we acquire their tracking results.

**Merge/Split condition.** (a) In frame 5427, we detect that A and B are merging. (b) We track A and B as one target. After 55 frames, we detect that they have split. (c) Some random image patches are sampled in them. (e) We use their classifiers to obtain the score maps. (f) After the particle filtering process, we obtain the tracking results.

The observation model to be updated is

$$P_{\text{video}}(z_t|x_{k,t}) = \frac{1}{\sqrt{2\pi}/\sigma} \sum_{i=1}^{N} \beta_i \exp\left(-\frac{(d(x_{k,t}) - d^i_{k,t})^2}{\sigma^2}\right) \tag{11}$$

where $\beta_i = \frac{I_i}{\sum_{i=1}^{N} I_i}$ is the normalized score of image patch $i$, and $I$ is the score value. Function $d(\cdot)$ converts the “two feet position” and the persons’ state into the center position of the bounding box in image coordinates, which are dependent on the extrinsic calibration parameters. Hence, $d(x_{k,t})$ is the center position of candidate target $k$ to be estimated, $d^i_{k,t}$ is the center position of image patch $i$, and $\sigma$ is the covariance which depends on the size of the image patch. For each target, the observation is further peaked around its real region, and the particles sampled in this region will have higher weights. As a result, the particles are much more focused around the true target state after each level’s re-weighting and re-sampling. Finally, we utilize the weighted average values of these particles as the estimated position of the person, and this was usually close to the real center position of the person’s bounding box. Subsequently, we can easily obtain the new position of these interacting targets and avoid the “drift problem”. An overview of the process is shown in Fig. 11.
The first row shows the results using our system, and the second and the third rows show the results of laser-based PF trackers and vision-based PD-PF trackers. Please note the key targets A, B, C, D and E (E appears in frame 17707). Targets A and C, A and E are merging in frame 17707 and 17789 respectively. 42 frames later (frame 17831), only our tracking system has correctly maintained their trajectories. For more results of our system, please see our supplementary video.

Table I. Successful Completion of Tracking in Challenging Situations

<table>
<thead>
<tr>
<th></th>
<th>Correlated Targets</th>
<th>Merge/Split</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Success/Total</td>
<td>Success Rate</td>
</tr>
<tr>
<td><strong>Our System</strong></td>
<td>3396/3832</td>
<td>88.62%</td>
</tr>
<tr>
<td>Laser-based PF Tracker</td>
<td>1821/3832</td>
<td>47.52%</td>
</tr>
<tr>
<td>Vision-based PD-PF Tracker</td>
<td>2566/3832</td>
<td>66.96%</td>
</tr>
</tbody>
</table>

In the non-correlated condition, we utilize the classifiers of these targets to identify them (as shown in Fig. 12). Hence, we can link the trajectories of these targets without difficulty.

With the help of the classifiers and visual information, our method is able to deal with various complex tracking situations. In addition, the tracking and learning complements the laser and vision in the proposed method, consequently becoming an adaptive loop that ensures the entire process can be completely online and automatic.

7. EXPERIMENTS AND RESULTS

We evaluated our tracking system in a real scene at a library. In order to deal with occluded targets, we utilized two single-row laser scanners. In addition, one camera was set on the third floor of the building. The test lasted for 30 min (about 45000 frames). When the distance between targets was less than 1.5 m, we started to train the classifiers, and once the distance between them was less than 0.3 m, we considered it to be a challenging situation. In this section, we will present our tracking results and perform some quantitative evaluations.

7.1. Tracking Results

Fig. 13 shows an example of the tracking results. The first row contains our tracking results, the second contains the results with the laser-based tracker, and the third shows the results with the visual
Quantitative evaluation of tracking situation switch module. The first row (ground truth) shows the number of correlated targets, merge/split targets, and appearing/disappearing targets in 5000 frames (the vertical axis shows the target number). The second row shows the corresponding number of detections and false alarms given by our system in these frames.

Table II. Detection Accuracy

<table>
<thead>
<tr>
<th></th>
<th>Detection Rate</th>
<th>False Alarm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correlated Targets</td>
<td>93.35%</td>
<td>2.35%</td>
</tr>
<tr>
<td>Merge/Split Targets</td>
<td>89.27%</td>
<td>3.27%</td>
</tr>
<tr>
<td>Appearing/Disappearing</td>
<td>95.26%</td>
<td>6.73%</td>
</tr>
</tbody>
</table>

We selected 6000 continuous frames in which interactions frequently took place, and performed a statistical survey of how many challenging situations (such as correlated targets or “merge/split” conditions) we could deal with. Once one of these conditions occurred with no failure in tracking, a successful completion was counted. The details of this survey are shown in Table I. From this table, we can see that our tracking system can deal with most correlated targets or merge/split conditions, which are difficult for the traditional laser-based or visual-based trackers.

7.2. Quantitative Evaluation of Tracking Situation Switch Module

The tracking situation switch module is a key component in the overall system, and can greatly influence its performance. Hence, we conducted a quantitative evaluation of its accuracy. We performed a statistical survey of 5000 continuous frames, and the details are shown in Fig. 14. The first row (ground truth) shows the number of correlated targets, merge/split targets, and appearing/disappearing targets in the 5000 frames, which were obtained in a semi-automatic way (system detection + manual labeling). In this experiment, we invited two people to perform the manual labeling, one from an academic background and the other with no academic background in this area. The second row of Fig. 14 shows the number of detections and false alarms in these frames. From this figure, we can see that this module achieves a high accuracy and can easily detect various tracking situations, which ensures the overall system automatically switches between tracking and learning. The overall detection accuracy of this module is shown in Table II.
Fig. 15. Quantitative evaluation of online learning: This comparison was conducted between the trackers without online learning and our system (with online learning). Fig.(a) shows the correctly tracked targets by the two systems over 5000 frames. Fig.(b) shows actual target number and interactions in these frames.

### Table III. Success Rate and Failure Tracking Situations of Two Methods

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Success Rate</th>
<th>Missed Targets</th>
<th>False Locations</th>
<th>Identity Switches</th>
</tr>
</thead>
<tbody>
<tr>
<td>PF Trackers (without online learning)</td>
<td>76.37%</td>
<td>46.23%</td>
<td>19.25%</td>
<td>34.52%</td>
</tr>
<tr>
<td><strong>Our System</strong> (with online learning)</td>
<td><strong>92.23%</strong></td>
<td>73.26%</td>
<td>11.35%</td>
<td>15.39%</td>
</tr>
</tbody>
</table>

### 7.3. Quantitative Evaluation of Online Learning

Online learning is a key technique in the proposed system, as it can help the overall system to deal with some challenging situations. Hence, we conducted a quantitative comparison to show the importance of the online learning to the overall system. The quantitative comparison was conducted between two methods: our system and laser-based PF trackers (without online learning and as described in Section 5). We performed the comparison on a continuous 5000 frames, and the ground truth was obtained in a semi-automatic way (trackers + manual labeling). In this experiment, we also invited the same two people to perform the manual labeling. By comparing the tracking results with the ground truth, it is easy for us to recognize different failed tracking situations, including missed targets, false locations, and identity switches. For more details about these, please refer to our previous work [Song et al. 2008b]. The details of this comparison are illustrated in Fig. 15. From this figure, we can see that, with the help of online learning, our system could obtain more robust tracking results than laser-based PF trackers when interactions occurred (as shown in frames 1386 and 3383). The overall success rate and failure tracking situations of these methods is shown in Table III, and the online learning is found to provide a 15% performance improvement over the PF trackers without it.

### 7.4. Quantitative Comparison and System Evaluation

We evaluated the performance of our tracking system from two aspects: tracking accuracy and time-cost (CPU computational time), and conducted a quantitative comparison among three systems: laser-based only [Song et al. 2008], camera-based only [Song et al. 2008b], and the system proposed in this paper. All of these systems have online learning components. We performed a statistical survey of 5000 continuous frames, the details of which are shown in Fig. 16. Please note that the ground truth was obtained by a semi-automatic method (trackers + manual labeling). By comparing the tracking results with the ground truth, it was easy for us to recognize different failed tracking...
Fig. 16. Quantitative comparison and system evaluation. (a) Correctly tracked targets of three systems over 5000 frames. (b) Actual target and interaction numbers in these frames. (c) Time-cost factors of three systems per frame (note that the time-cost factor is normalized to $0 - 1$).

Table IV. Overall Tracking Performance

<table>
<thead>
<tr>
<th>System</th>
<th>Average Accuracy</th>
<th>Average Time-cost</th>
<th>Missed Targets</th>
<th>False Locations</th>
<th>Identity Switches</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our System</td>
<td>92.23%</td>
<td>0.1963</td>
<td>73.26%</td>
<td>11.35%</td>
<td>15.39%</td>
</tr>
<tr>
<td>Laser Only</td>
<td>81.36%</td>
<td>0.0773</td>
<td>57.38%</td>
<td>16.29%</td>
<td>26.33%</td>
</tr>
<tr>
<td>Vision Only</td>
<td>85.82%</td>
<td>0.7223</td>
<td>46.36%</td>
<td>29.86%</td>
<td>23.78%</td>
</tr>
</tbody>
</table>

situations, including missed targets, false locations, and identity switches. In addition, the time-cost factor was normalized to the range 0-1. Moreover, once we start to train the classifiers, the interactions should also be counted in this experiment.

From Fig. 16, we can see that the time-cost factor of our system was the same as that of the laser-based one (without visual processing) when there was no interaction, and much more efficient than the vision-based system. When interactions occurred, although our system needed some extra computation (for the visual processing), it could deal with most interactions, and performed better than the laser-based system. In addition, the time-cost factor of our system was still less than that of the vision-based one, because only a limited number of targets required visual processing. The overall tracking accuracy and time-cost factor are shown in Table IV. In conditions of an average 10.18 targets per frame and 2.17 interactions per frame, our system obtained the highest tracking accuracy and was much faster than the vision-based system. Indeed, as the number of tracking targets and interactions increases, our system exhibits a much better performance than the other two systems.

7.5. Failed Tracking Caused by Multiple Kinds of Objects

The proposed system can maintain correct tracking in most situations. However, some tracking failures occasionally took place when some other kinds of targets appeared, such as bicycles and cars. Because the “two feet walking” model cannot apply to these targets, the tracking of these targets usually failed, significantly decreasing the performance of the proposed system. Some examples are shown in Fig. 17 (first row).

In order to deal with these problems, we add a classification module to our system. Some geometric and motion features are extracted from the laser data (for instance, people’s data is usually like a circle, and a bicycle is like a line), and the off-line trained AdaBoost classifier is utilized to perform the classification. Some results are shown in Fig. 17 (second row).
After the classification, we utilize different motion models for different targets (the “two feet walking” model for people, and the constant velocity model for bicycles). With the help of the classification module, the tracking failures caused by multiple object types are minimized, and the performance of the proposed system is significantly improved. Some results are shown in Fig. 17 (third row).

8. CONCLUSION

In this paper, we presented a novel multi-target tracking system using lasers and vision in a wide and open area. By combining the lasers and vision with tracking and learning, the proposed system easily dealt with various challenging tracking situations in the tracking. In addition, experimental results showed that our system fully incorporated the respective advantages of the two sensors and obtained a superior performance in both tracking accuracy and time-cost factor.

Sometimes, the performance of the proposed system was degraded by multiple kinds of objects (such as pedestrians, bicycles, and cars). By adding the classification module, this problem was easily solved. Meanwhile, the proposed system will encounter some problems while some persons who have the similar appearances are walking together, and this is the potential limitation of our system. At present, EOH feature can in part deal with this problem. However, if the appearance of some interacting persons are almost same, our system will also encounter some uncertainty in labeling. In the future, some more powerful visual feature should be explored to deal with this problem. On the other hand, some semantic scene knowledge learning module should be combined into our system either, which can further improve the tracking and classification results.
ACKNOWLEDGEMENT

This work was partially supported by a Grant-in-Aid for Young Scientists (23700192) and “Strategic Project to Support the Formation of Research Bases at Private Universities”: Matching Fund Subsidy by Japan’s Ministry of Education, Culture, Sports, Science, and Technology. This work was also partially funded by NSFC Grants (No.60975061) of China and Microsoft Research. We thank all anonymous reviewers for their helpful comments on this work.

REFERENCES


