Proceedings of the
2007 IEEE Intelligent Vehicles Symposium
Istanbul, Turkey, June 13-15, 2007

Detection and Tracking of Multiple Pedestrians in Automotive Applications

Richard Arndt*, Roland Schweiger†, Werner Ritter‡, Dietrich Paulus* and Otto Löhlein‡
*University of Koblenz-Landau, Institute for Computational Visualistics, 56070 Koblenz, Germany
rarndt|paulus@uni-koblenz.de
‡DaimlerChrysler AG, Research & Technology, Department GR/EAP, 89069 Ulm, Germany
werner.r.ritter|otto.loehlein@daimlerchrysler.com
†University of Ulm, Institute for Measurement Control and Microtechnology, 89069 Ulm, Germany
uni-ulm.schweiger@daimlerchrysler.com

***Abstract*— We present a method for tracking an unknown and changing number of far away pedestrians in a video stream. Multiple particle filter instances are utilized which track single pedestrians independently from each other. The tracking is guided by a cascade classifier which is integrated into the particle filter framework. In order to be able to detect hardly visible pedestrians and to filter out isolated false positives of the classifier, we developed a detection criterion for particle filters which follows the track-before-detect paradigm. The system nearly works in real time.**

I. INTRODUCTION

Pedestrians are the weakest traffic participants because they are hardly protected against the consequences of an accident. For that reason it is required to implement active safety systems which are capable of detecting pedestrians in order to warn the driver in case of an accident risk. It is of eminent importance that pedestrians are timely detected at night, as the number of fatal accidents is considerably higher at night time than at day time whereas the traffic density is substantially lower. As non-warning night vision systems are already available, the development of intelligent systems which detect and track pedestrians in video streams is currently investigated.

A. Related work

The integration of a cascade classifier (CC) into a particle filter (PF) based framework for tracking multiple persons has been also carried out in [6]. There the classifier is utilized for implementing a proposal density [1] which takes the current video frame into account.

In [10] and [13] support vector machines are employed for detecting pedestrians in far infrared (FIR) night vision images. Subsequent tracking using one Kalman filter for each detection is performed in [13].

In addition to pixel-based approaches for detecting pedestrians there also exist systems which rely on shape-based features. In [3] a system which is capable of tracking the shape and the coordinate of several pedestrians is introduced.

The presented work is a result of a diploma thesis at department GR/EAP, DaimlerChrysler AG Research & Technology, Germany, 89069 Ulm.

A similar approach which utilizes B-Spline curves for the approximation of pedestrian shapes is presented in [8]. As our system should be capable of tracking far away pedestrians, we do not utilize shape based features which are different to obtain for far away, i.e. small, objects.

B. Structure of the paper

We briefly introduce our tracking system in Sec. II before we provide a detailed description of it in Sec. III, IV and V. The evaluation results are presented in Sec. VI. Finally, we draw a conclusion and outline future work in Sec. VII.

II. SYSTEM DESIGN

A rough overview of our multi-target tracking system is given in Fig. 1. The system is capable of tracking several pedestrians through the recursive probabilistic filtering of a monocular stream of near infrared (NIR) images. Under the assumption that individual pedestrians move independently from each other, we employ n_{PF} parallel working PF

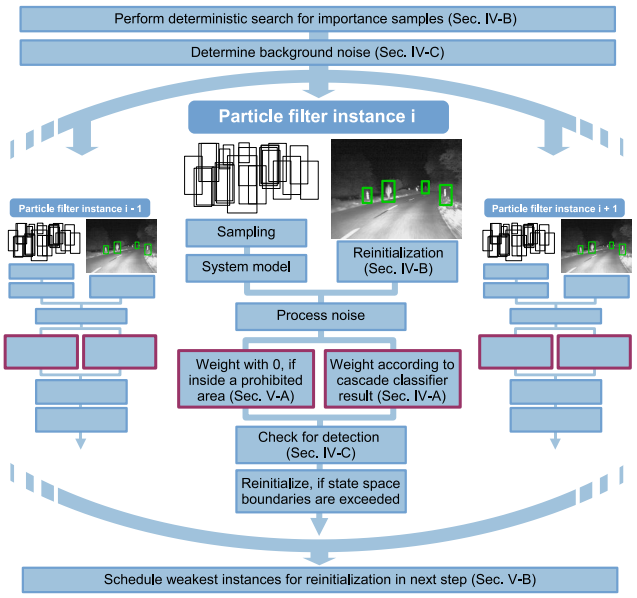

Fig. 1. One time step of the multi-target tracking system.

instances which track one object respectively. In Sec. IV we therefore firstly describe how a single pedestrian is tracked using one PF. The procedure is guided by a CC (see Sec. III) which is trained to separate image regions into pedestrians and background. It is integrated into the PF through the definition of a function for weighting particles (see Sec. IV-A). The goal of our efforts is the enhancement of the detection performance of the CC through recursive filtering of its classification results. We will see in Sec. IV-B and IV-C that the classifier can be also used for the definition of a system state prior which takes the current observation into account and a detection criterion which follows the track-before-detect (TBD) paradigm [11]. The extensions which are needed for the parallel execution of multiple PF instances are detailed in Sec. V.

III. CASCADE CLASSIFIER

Viola and Jones have shown in [12] that the detection of complex objects can be carried out in real time using a cascaded classifier which is capable of separating image regions into two classes[1] utilizing a chain of classifier stages. With increasing stage, the complexity of these classifiers increases. The simple classifiers in the lower stages discard the majority of image regions which do not show the learned class while the remaining are separated out by the sophisticated higher stages of the CC. An image region is represented through a rectangular search window $\boldsymbol{w} \in \mathbb{N}_0^4$ which is defined by its upper left and lower right points and features a fixed width to height ratio r_{SW}. Each of the n_c classifier stages applies a set of Haar Wavelet features [7] in order to decide whether a search window $\boldsymbol{w}$ belongs to the learned class or not. In the former case, the window is passed to the next classifier stage.

We denote the cascaded classification of a search window $\boldsymbol{w}$ by the application of the function $c : \mathbb{N}_0^4 \mapsto [0, n_c] \subset \mathbb{N}_0$ which returns the number ν of the classifier stage which has been passed by the search window before it was discarded:

$$\nu = c\left(\boldsymbol{w}\right) \quad . \tag{1}$$

The window is classified as an object if it passes the detection stage ν_{D}, i.e. if $c\left(\boldsymbol{w}\right) \geq \nu_{\mathrm{D}}$.

IV. PARTICLE FILTER

Object detection and tracking is carried out using particle filters [1]. Through recursive probabilistic filtering of the incoming monocular NIR image stream, the state vector

$$\boldsymbol{q}_t = \left(\alpha, x, \dot{x}, y, \dot{y}\right)^{\mathrm{T}} \in \mathbb{R}^5 \tag{2}$$

of a pedestrian is estimated for each time step t. Due to the application of the ground plane assumption, only the longitudinal (x) and the lateral (y) coordinate of a pedestrian with respect to the car coordinate system need to be estimated. The corresponding velocity vector $(\dot{x}, \dot{y})$ is estimated relatively to the pedestrians coordinate as we include the ego-motion of the vehicle in the pedestrians first order system model. Furthermore, the tilt angle α of the car is included into the state vector in order to be able to track also far away pedestrians which are moved far below or above the ground plane through the pitch movements of the vehicle.

The Gaussian process noise of the system model is described by the five standard deviations σ_α, $\sigma_{\dot{x}}$, $\sigma_{\dot{y}}$, $\sigma_x\left(x; \sigma_{\mathrm{IM}}\right)$ and $\sigma_y\left(x; \sigma_{\mathrm{IM}}\right)$. The first three are assumed to be fixed and affect the tilt angle α and the velocity vector $(\dot{x}, \dot{y})$ respectively. The latter two depend on the standard deviation σ_{IM} which defines a Gaussian process noise in the image plane. Depending on the current x coordinate and the geometry of the camera, $\sigma_x\left(x; \sigma_{\mathrm{IM}}\right)$ and $\sigma_y\left(x; \sigma_{\mathrm{IM}}\right)$ are computed in a way that the standard deviation of the projected position vector (x, y) is approximately σ_{IM} pixel in the horizontal and vertical direction of the image plane.[2]

Utilizing the PF framework, the distribution $p\left(\boldsymbol{q}_t | \langle \boldsymbol{o} \rangle_t\right)$ of the state vector $\boldsymbol{q}_t$ is estimated through a set $\mathfrak{S}_t = \{\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_{n_{\mathrm{P}}}\}$ of n_{P} particles. Each particle $\boldsymbol{\xi}_i = \left(\boldsymbol{s}_i, \tilde{w}_t\left(\boldsymbol{s}_i\right)\right)$ consists of a hypothesis $\boldsymbol{s}_i \in \mathbb{R}^5$ of the true state vector $\boldsymbol{q}_t$ and a corresponding weight $\tilde{w}_t\left(\boldsymbol{s}_i\right) \in [0, 1] \subset \mathbb{R}$. As $\mathfrak{S}_t$ approximates a distribution, the sum of the weights is normalized. The distribution $p\left(\boldsymbol{q}_t | \langle \boldsymbol{o} \rangle_t\right)$ is estimated recursively based on the incoming observations $\boldsymbol{o}_0, \ldots, \boldsymbol{o}_t = \langle \boldsymbol{o} \rangle_t$, i.e. the frames of the NIR camera. An estimation $\hat{\boldsymbol{q}}_{t+1}$ of the current state vector can be carried out by updating the last approximation of $p\left(\boldsymbol{q}_t | \langle \boldsymbol{o} \rangle_t\right)$, i.e. $\mathfrak{S}_t$, with respect to the current observation $\boldsymbol{o}_{t+1}$. One PF cycle[3] consists of sampling a new particle set from $\mathfrak{S}_t$, moving each particle in state space by applying the system model as well as the corresponding process noise and finally validating each particle against $\boldsymbol{o}_{t+1}$ by applying a function $g\left(\boldsymbol{q}_{t+1}\right)$ which assigns a new weight to the particle (see also Fig.1). The resulting particle set $\mathfrak{S}_{t+1}$ is an approximation of $p\left(\boldsymbol{q}_{t+1} | \langle \boldsymbol{o} \rangle_t, \boldsymbol{o}_{t+1}\right)$ and can be utilized for deriving the desired state estimate $\hat{\boldsymbol{q}}_{t+1}$.

A. Particle Weighting

In order to compare a state hypothesis with the true state of a pedestrian which is captured by the current NIR image $\boldsymbol{o}_{t+1}$, it is required to project hypotheses from the car coordinate system into the image coordinate system. The function $\chi : \mathbb{R}^5 \mapsto \mathbb{N}_0^4$

$$\boldsymbol{w} = \chi\left(\boldsymbol{q}_{t+1}\right) \tag{3}$$

maps a hypothesis $\boldsymbol{q}_{t+1}$ to a rectangular search window $\boldsymbol{w}$ which defines a sub-image of $\boldsymbol{o}_{t+1}$. A pinhole camera model is utilized for the projection of (x, y) into the image plane following the current tilt α of the particle. The spatial extend of $\boldsymbol{w}$ is determined by using a fixed pedestrian height of 1.8 m and the CC's aspect ration r_{SW} for calculating its width.

In order to combine the CC with a PF, we developed a function $g_{\mathrm{CAS}} : \mathbb{R}^5 \mapsto [0, 1] \subset \mathbb{R}$ which assigns a weight to

[1] We will refer to the two classes as 1) object or pedestrian and 2) background.

[2] An exact conversion of Gaussian noise from state into image space is impossible due to the perspective projection of the camera.

[3] The cycle is described for a CONDENSATION PF [5].

a particle that depends on the classification of the corresponding search window $\boldsymbol{w}$. As the classification result of the classifier is a stochastic process, the weight function g_{CAS} has to take this into account by including an appropriate observation noise. The determination of it required an evaluation of the CC w.r.t. the correlation between the conformance of a search window $\boldsymbol{w}$ with a pedestrian and its classification result, i.e. the reached classifier stage $c(\boldsymbol{w})$.

1) Evaluation of the Cascade Classifier: The evaluation has been carried out using 14,283 NIR images of a ground truth image data base. Pedestrians are labeled with bounding boxes on these images. The conformance between ground truth labels and search windows was measured by their coverage. The function $\mathrm{cov} : \mathbb{N}_0^4 \times \mathbb{N}_0^4 \mapsto [0,1] \subset \mathbb{R}$

$$\mathrm{cov}(\boldsymbol{a}, \boldsymbol{b}) = \frac{\boldsymbol{a} \cap \boldsymbol{b}}{\boldsymbol{a} \cup \boldsymbol{b}} \tag{4}$$

was applied to determine the coverage of two rectangular search windows $\boldsymbol{a}$ and $\boldsymbol{b}$.

A search window generator was used to create a very dense set of 3,179,134 evenly distributed windows. The CC has been applied to each of them for every data base image. According to its coverage with a ground truth label and its reached classifier stage, each search window with at least 1% ground truth coverage has been counted in a 2D-histogram. Three slices of it are shown in Fig. 2. It is evident, that windows which reach high classifier stages show higher coverage with pedestrians.

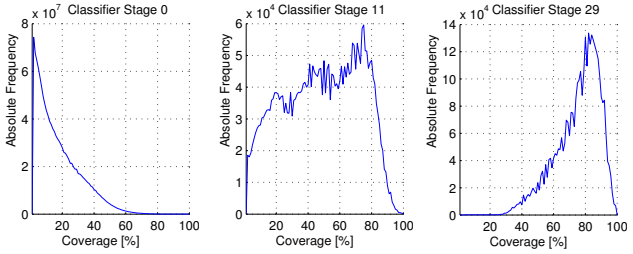

Fig. 2. Correlation between pedestrian coverage and classification result of a search window. Depicted are the histograms for windows $\boldsymbol{w}$ with $c(\boldsymbol{w}) = \{0, 11, 29\}$. The classifier consists of 29 stages. The different scales of the histograms demonstrate the functionality of the CC. Most of the search windows are discarded in the lower stages. The dense set of search windows that has been evaluated per image explains the high absolute frequency of windows that feature a large coverage with pedestrian labels.

2) Weighting through Conclusion of Coverage: The histograms which are shown in Fig. 2 already cover the uncertainty of the CC. Therefore we decided to convert them into distributions and use them for designing a weight function g_{CAS} which assigns a coverage $\gamma \in [0,1] \subset \mathbb{R}$ with a pedestrian to a particle. In the following we will refer to the coverage distribution of the ν-th stage by $p_\nu^{\mathrm{cov}}(\gamma)$. Following this notation, the upper left histogram of Fig. 2 corresponds to $p_0^{\mathrm{cov}}(\gamma)$ while the lower right corresponds to $p_{29}^{\mathrm{cov}}(\gamma)$.

Calculating the (unnormalized) weight of a state hypothesis $\boldsymbol{q}_{t+1}$ first requires its classification in image space using the CC. The classifier stage ν of the search window $\chi(\boldsymbol{q}_{t+1})$ is subsequently used to calculate the particle weight by sampling a coverage γ from the corresponding distribution $p_\nu^{\mathrm{cov}}(\gamma)$:[4]

$$g_{\mathrm{CAS}}(\boldsymbol{q}_{t+1}) \sim p_\nu^{\mathrm{cov}}(\gamma) \mid \nu = c(\chi(\boldsymbol{q}_{t+1})) \quad . \tag{5}$$

Due to the usage of a stochastic weight function we employ a minimum mean square error (MMSE) estimate to predict the state vector of a pedestrian:

$$\hat{\boldsymbol{q}}_{t+1}^{\mathrm{MMSE}} \approx \sum_{i=1}^{n_{\mathrm{P}}} \boldsymbol{s}_i \tilde{w}_{t+1}(\boldsymbol{s}_i) \quad . \tag{6}$$

B. Importance A Priori Sampling

The application of a PF for the recursive estimation of the state vector $\boldsymbol{q}_{t+1}$ requires the existence of a prior $p(\boldsymbol{q}_0)$ which generates hypotheses of the initial state of the observed system [1]. As this distribution is usually unknown, many tracking applications employ a random initialization of the particle set (e.g. [5] or [4]). However, this approach is unfeasible for the tracking of far away pedestrians. Therefore, we developed an importance a priori distribution $p^{\mathfrak{I}}(\boldsymbol{q}_0)$ which takes the current image into account and allows a targeted initialization of a particle set in image space.

We perform a deterministic search in image space in order to determine good starting points $\boldsymbol{s}_j^{\mathfrak{I}}$ in state space for the subsequent tracking. We generate these start points which we will call importance samples in the following from the $n_{\mathfrak{I}}$ best rated search windows $\boldsymbol{w}_j^{\mathfrak{I}}$ of the current frame. The CC is employed to select them from an evenly but coarse distributed search window set $W^{\mathfrak{I}}$ which is generated utilizing the ground plane assumption and the geometry of the camera. By slightly relaxing the ground plane assumption, it is possible to deduce a coordinate $\left(x_j^{\mathfrak{I}}, y_j^{\mathfrak{I}}\right)$ as well as a tilt $\alpha_j^{\mathfrak{I}}$ from each search windows $\boldsymbol{w}_j^{\mathfrak{I}}$.

The initialization of a PF instance at time step 0 is carried out using the importance sample $\boldsymbol{s}_j^{\mathfrak{I}}$ which corresponds to the highest rated search window $\boldsymbol{w}_j^{\mathfrak{I}}$ that has not already been used for the initialization of another instance. The state hypothesis $\boldsymbol{s}_i$ of each particle $\boldsymbol{\xi}_i$ is drawn from $p^{\mathfrak{I}}(\boldsymbol{q}_0)$ by adding a random velocity and a sample of the process noise to $\boldsymbol{s}_j^{\mathfrak{I}}$.

Evaluations of the CC have shown that pedestrians usually cause multiple detections in a fine grained grid of search windows. Thus, a coarse search is sufficient for determining the importance samples $\boldsymbol{s}_j^{\mathfrak{I}}$.

C. Robust Detection and Tracking

We developed a detection criterion that follows the TBD paradigm in order to enable the tracking of weakly distinct pedestrians. Furthermore, the tracking of isolated false positives of the CC is pruned by the application of this paradigm as the ability to predict the movement of a pedestrian is explicitly considered in the criterion. The usage of multiple PF instances favors the implementation of the TBD paradigm. Since not all instances are required for tracking pedestrians, it is affordable to let some instances track weak targets which might become detections in the next few time steps.

[4]The transformation method [9] is used for the sampling.

We follow the approach of [2] for integrating the TBD paradigm into the PF framework. They define a detection criterion which is based on the likelihood ratio

$$\lambda\left(\boldsymbol{o}_{t+1}\right)=\frac{p\left(\boldsymbol{o}_{t+1}|\mathcal{H}_1\right)}{p\left(\boldsymbol{o}_{t+1}|\mathcal{H}_0\right)}\in\mathbb{R} \quad . \tag{7}$$

It is computed by determining the probabilities $p\left(\boldsymbol{o}_{t+1}|\mathcal{H}_1\right)$ and $p\left(\boldsymbol{o}_{t+1}|\mathcal{H}_0\right)$. The former expresses the likeliness of an observation $\boldsymbol{o}_{t+1}$ with the hypothesis $\mathcal{H}_1$, which stands for the assumption that $\boldsymbol{o}_{t+1}$ is caused by a target. The latter expresses analogously the likeliness of $\mathcal{H}_0$ which states that $\boldsymbol{o}_{t+1}$ contains background or noise. The likelihood ratio implements a signal to noise ratio. A detection is reported by the tracking system if $\lambda\left(\boldsymbol{o}_{t+1}\right)$ exceeds the threshold $\theta_\lambda\in\left]1,\infty\right]\subset\mathbb{R}$.

In [2] it is shown that $p\left(\boldsymbol{o}_{t+1}|\mathcal{H}_1\right)$ can be approximated by a set of particles by calculating the mean of all unnormalized particle weights. As we use the function g_{CAS} for determining these weights, $p\left(\boldsymbol{o}_{t+1}|\mathcal{H}_1\right)$ is approximated by

$$p\left(\boldsymbol{o}_{t+1}|\mathcal{H}_1\right)\approx\frac{1}{n_{\mathrm{P}}}\sum_{i=1}^{n_{\mathrm{P}}}g_{\mathrm{CAS}}\left(\boldsymbol{s}_i\right) \quad . \tag{8}$$

The calculation of $p\left(\boldsymbol{o}_{t+1}|\mathcal{H}_0\right)$ is not detailed in [2] as they apply the likelihood ratio based detection criterion to track targets on a radar screen which has a well known background noise. In order to apply the likelihood ratio criterion to vision based tracking, we developed a method which utilizes the weight function g_{CAS} for the approximation of $p\left(\boldsymbol{o}_{t+1}|\mathcal{H}_0\right)$. The idea behind our approach is as follows: As the estimation of $p\left(\boldsymbol{o}_{t+1}|\mathcal{H}_1\right)$ is carried out using the mean weight of hypotheses which are assumed to contain a pedestrian, we apply an analogous approach for the estimation of $p\left(\boldsymbol{o}_{t+1}|\mathcal{H}_0\right)$ by calculating the mean weight of hypotheses which are assumed to show background. A set of search windows which are evenly distributed across the image is a set of background hypotheses as in the majority of cases these windows will contain background. Therefore, we re-use the search window set $W^{\mathfrak{I}}$ (see Sec. IV-B) to approximate the background noise by

$$p\left(\boldsymbol{o}_{t+1}|\mathcal{H}_0\right)\approx\frac{1}{|W^{\mathfrak{I}}|}\sum_{\boldsymbol{w}\in W^{\mathfrak{I}}}g'_{\mathrm{CAS}}\left(\boldsymbol{w}\right) \quad . \tag{9}$$

The function $g'_{\mathrm{CAS}}:\mathbb{N}_0^4\mapsto[0,1]\subset\mathbb{R}$ denotes the weighting of a search window $\boldsymbol{w}$ analogously to (5):

$$g'_{\mathrm{CAS}}\left(\boldsymbol{w}\right)\sim p_\nu^{\mathrm{cov}}\left(\gamma\right)\ |\ \nu=c\left(\boldsymbol{w}\right) \quad . \tag{10}$$

Determining an appropriate value for the threshold θ_λ is a hard task. On the one hand it has to be low enough to allow the detection and tracking of weak targets while on the other hand, it has to be high enough in order to prevent the system from tracking false positives of the CC. In order to solve this problem, we employ the low-pass filtered likelihood ratio

$$\tilde{\lambda}\left(\boldsymbol{o}_{t+1}\right)=\left(1-\delta_\lambda\right)\lambda\left(\boldsymbol{o}_{t+1}\right)+\delta_\lambda\tilde{\lambda}\left(\boldsymbol{o}_t\right) \tag{11}$$

to decide whether a PF tracks a pedestrian or a background object. The ratio $\tilde{\lambda}\left(\boldsymbol{o}_{t+1}\right)$ depends on the low-pass factor $\delta_\lambda\in[0,1]\subset\mathbb{R}$. A detection is reported if

$$\tilde{\lambda}\left(\boldsymbol{o}_{t+1}\right)>\theta_\lambda \tag{12}$$

holds. The tracking lasts until $\tilde{\lambda}\left(\boldsymbol{o}_{t+1}\right)$ drops below θ_λ or the pedestrian leaves the boundaries of the relevant partition of the state space. In the latter case the PF instance is reinitialized using the prior $p^{\mathfrak{I}}\left(\boldsymbol{q}_0\right)$.

Using the low-pass filtered likelihood ratio implies that pedestrians are not detected immediately. The system rather concludes their presence by a series of high unfiltered likelihood ratios. Thus, the threshold θ_λ can be set to a considerably lower value without allowing the system to track isolated false positives. Furthermore, a lower detection threshold allows the compensation of minor errors during the tracking of a pedestrian.

It is crucial for a successful application of this detection criterion that a proper initialization value for $\tilde{\lambda}\left(\boldsymbol{o}_{-1}\right)$ is chosen before the first observation $\boldsymbol{o}_0$ is utilized for weighting the new particles which have been drawn from $p^{\mathfrak{I}}\left(\boldsymbol{q}_0\right)$. Due to the fact that $p^{\mathfrak{I}}\left(\boldsymbol{q}_0\right)$ generates samples nearby potential pedestrians, i.e. regions which are highly rated by the CC, it is impossible to initialize $\tilde{\lambda}\left(\boldsymbol{o}_{-1}\right)$ with the first measured likelihood ratio $\lambda\left(\boldsymbol{o}_0\right)$. For that reason, we use a fixed value for initializing the low-pass filtered likelihood in between the two hypotheses $\mathcal{H}_0$ and $\mathcal{H}_1$:

$$\tilde{\lambda}\left(\boldsymbol{o}_{-1}\right)=1+\frac{\theta_\lambda-1}{2} \quad . \tag{13}$$

V. MULTI-INSTANCE PARTICLE FILTER

We use several PF instances to track multiple pedestrians simultaneously, following the multi-instance particle filter (MIPF) approach which is presented in [4]. In order to enable a stable tracking of several targets using parallel working PF instances, it is necessary to ensure that one object is only tracked by at most one instance. The authors introduce prohibited areas in state space and a fixed ranking of the instances in order to prevent multiple instances from tracking the same object. Prohibited areas are declared by instances which are tracking a detection. If a particle enters the prohibited area of another instance which has a higher rank, it is weighted with 0. In Sec. V-A we introduce a criterion for dynamically ranking PF instances which is mainly based on prohibited areas that are defined in image space.

As the detection and the tracking of objects is carried out simultaneously by using multiple PF instances, it is required to ensure that the entire state space is continuously searched for objects. Following [4], we reinitialize a fixed number of n_{RI} instances after each time step in order to keep the system searching for new objects. These instances are chosen according to a reinitialization criterion which we will introduce in Sec. V-B.

A. Ranking of Instances

The PF instances are processed in a fixed order each time step. In contrast to [4], we do not rank the instances according to this order. Instead, we always prioritize instances which track a detection. These instances define prohibited

areas in order to prevent other instances from tracking the same pedestrian.

Measuring the distance between the MMSE state estimate of a detected pedestrian and a state hypothesis is required in order to decide whether the particles of a subordinate instance are inside a prohibited area. As the coordinate (x, y) of a pedestrian cannot be estimated precisely enough using a monocular night vision device we employ the function $\mathrm{cov}_{\max} : \mathbb{N}_0^4 \times \mathbb{N}_0^4 \mapsto [0,1] \subset \mathbb{R}$ which measures the distance between two rectangular search windows $\boldsymbol{a}$ and $\boldsymbol{b}$ in image space by their maximum coverage:

$$\mathrm{cov}_{\max}(\boldsymbol{a}, \boldsymbol{b}) = \frac{\boldsymbol{a} \cap \boldsymbol{b}}{\min(\boldsymbol{a}, \boldsymbol{b})} \quad . \tag{14}$$

The term $\min(\boldsymbol{a}, \boldsymbol{b})$ denotes the smaller of the two search window areas. Using $\mathrm{cov}_{\max}$, a prohibited area is determined by defining a threshold $\theta_{\mathrm{V}} \in [0,1] \subset \mathbb{R}$ which stands for the maximum allowable coverage between the MMSE state estimate $\hat{\boldsymbol{q}}_{t+1}^{\mathrm{MMSE}}$ of a detected pedestrian and a state hypothesis $\boldsymbol{s}_i$ of a subordinate instance. Such a sample is weighted with 0, if

$$\mathrm{cov}_{\max}\left(\chi\left(\hat{\boldsymbol{q}}_{t+1}^{\mathrm{MMSE}}\right), \chi(\boldsymbol{s}_i)\right) \geq \theta_{\mathrm{V}} \quad . \tag{15}$$

In the case of a collision between two detections in image space, the longitudinal positions of the MMSE state estimates are used for determining the rank of the involved instances. The furthermost instance receives the lower rank in this situation.

B. Reinitialization Criterion

The effectual integration of the TBD paradigm into the MIPF framework requires the definition of an appropriate criterion for selecting the n_{RI} weakest PF instances at the end of each time step. The criterion has to be defined in a way that instances which have been tracking a weak object for a couple of time are preferred to instances that are tracking incoherent parts of the background. Furthermore, it is required that instances which are initialized nearby a promising object are allowed for superseding instances that are already tracking an undetected object.

Taking these requirements into account, the n_{RI} PF instances k which show the lowest reinitialization criterion

$$\rho_k = \max\left(\tilde{\lambda}_k(\boldsymbol{o}_{t+1}), \lambda_k(\boldsymbol{o}_{t+1})\right) \in \mathbb{R} \tag{16}$$

and are not tracking a detected pedestrian are scheduled for reinitialization in the next time step. The number k denotes the processing order of the instance.

VI. RESULTS

We used an INTEL P-IV 3.2 GHz machine for carrying out experiments on country roads. The images were captured using a NIR camera with a resolution of 640×480 pixels and 12 bit depth. A frame rate of approximately 15 fps was achieved using the parameter setup which is listed in Table I. With this configuration, the simultaneous tracking of $n_{\mathrm{PF}} - n_{\mathrm{RI}} = 8$ pedestrians is possible. Due to the fact that search windows which are discarded in high stages of the CC cause much more computational overhead than those which are discarded in low stages, the frame rate of the system considerably differs from the average rate if an image contains many pedestrians and / or false positives. Fig. 3 gives an impression of the system. The n_{PF} parallel running PF instance are separated by different colors. The detection of an instance, i.e. the MMSE state estimate, is depicted as a transparent box while its particles are represented by rectangular frames.

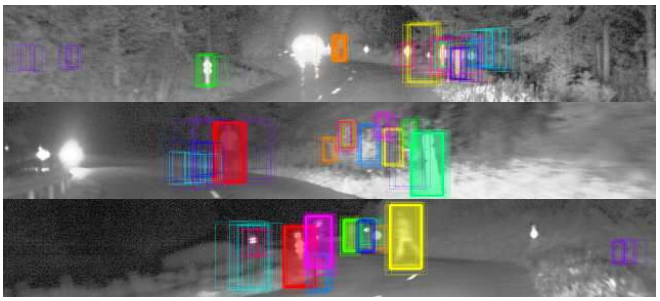

Fig. 3. Snapshots of the ground truth sequences. In frame one, a pedestrian (distance: > 100 m) is detected. A reflexion post and one pedestrian are detected in frame three. The fifth frame shows the detections of two pedestrians (green and yellow), one cyclist (purple) and one reflexion post.

The evaluation of the system has been carried out using six ground truth NIR sequences which have not been used for the training of the CC. For the evaluation, we only considered pedestrians which occur in between distances of 30 – 100 m, as a warning system for night view automotive applications should mainly detect obstacles which are beyond the cone of the low beam light. The remaining ones were ignored, i.e. they were neither counted as detections nor as false positives.

We performed the evaluation with two goals in mind. On the one hand, we wanted to analyze if our combination of a CC and a MIPF outperforms a system which only relies on the classifier. On the other hand, we wanted to determine appropriate assignments for the two main parameters of the tracking system, i.e. the detection threshold θ_λ (12) and the low-pass factor δ_λ (11). The remaining parameters of the PF system have been determined empirically during the experiments. They are listed in Table I.

TABLE I
ASSIGNMENTS OF PARAMETERS

n_c	29	n_{P}	100	σ_α [°]	0.04
n_{PF}	12	$W^\mathfrak{I}$	7738	$\sigma_{\dot{x}}$ / $\sigma_{\dot{y}}$ [m / s]	0.03
n_{RI}	4	θ_{V}	0.6	σ_{IM} [Pixel]	1.5

We compared the MIPF to a system which searches for pedestrians in a brute force (BF) way by classifying a very dense grid of 8,250,957 search windows per frame. We ran several variedly parametrized MIPF and BF systems simultaneously for the evaluation.

As the execution of the BF search is very time consuming, we decided to firstly determine an optimal value for δ_λ without taking the BF system into account. Therefore, several runs of our tracking system with different values for θ_λ and

δ_λ were performed. It turned out that independently from θ_λ the best performance is achieved if δ_λ takes values in between $[0.7, 0.8]$. This is a confirmation of the applicability of the TBD paradigm as the tracking of an object before its detection can be only enforced if δ_λ is assigned to a high value. The second evaluation has been carried out, using $\delta_\lambda = 0.8$.

In order to compare the performance of the BF system with our MIPF, we computed recall (R), precision (PR) and false alarm per image (FI) rates for differently parametrized BR and MIPF systems. The values of the former were generated by decreasing the detection threshold ν_D of the CC stepwise from 29 to 24 (see Sec. III) while the values of the latter were generated by running six MIPF systems which featured detection thresholds $\theta_\lambda \in \{1.1, 1.3, 1.5, 1.7, 1.9, 2.1\}$. A ground truth pedestrian was considered as being detected if its bounding box was covered by a detection of the BF system or the MIPF respectively by at least 20%. The coverage was determined using the function cov (4).

The results of the evaluation which are shown in Table II prove that the MIPF shows a considerably better performance than the BF system. Although, the number of false positives

TABLE II
EVALUATION RESULTS

BF	ν_D	24	25	26	27	28	29
	R	0.358	0.350	0.331	0.321	0.293	0.290
	PR	0.013	0.019	0.027	0.037	0.040	0.040
	FI	11.833	7.361	5.028	3.528	3.000	2.917
MIPF	θ_λ	1.1	1.3	1.5	1.7	1.9	2.1
	R	0.576	0.583	0.466	0.373	0.277	0.197
	PR	0.032	0.102	0.236	0.411	0.589	0.768
	FI	7.359	2.162	0.637	0.225	0.081	0.025

is clearly reduced, this advancement has to be considered carefully. As the BF system utilizes a very dense grid of search windows for detecting pedestrians, one erroneously classified object may cause many false positives. On the other hand, the MIPF will only report one false positive per frame for every erroneously detected object. However, a detailed analysis of the evaluation revealed that the MIPF indeed shows a higher precision than the BF search because isolated false positives of the CC are successfully filtered out by our system.

The analysis also revealed that due to our detection criterion the system is capable of tracking pedestrians over quite a long time. The ability of our system to compensate momentary weak responses from the CC resulted in uninterrupted tracks with a duration of more than 100 frames.

VII. CONCLUSION AND FUTURE WORK

We presented a system which is capable of detecting and tracking an unknown and changing number of pedestrians in a NIR image stream using a MIPF. Due to the integration of a detection criterion, which follows the TBD paradigm, a seamless transition between tracking and detection is achieved. This leads to a system which draws a conclusion about the existence of a pedestrian from tracking it. The system relies on a CC for validating state hypotheses in image space and for drawing samples from the system state prior. As the classification results of the classifier are imprecise, we developed a stochastic weighting function which explicitly takes the uncertainty of the results into account.

The evaluation results prove that our method shows a considerably better detection performance than a system which only relies on a CC. Nevertheless, the testing of the system revealed, that the pitch movements of the car nearly render impossible the tracking of pedestrians which occur at distances of 80 m and more. Therefore, a robust estimation of the pitch movement should be integrated into the system model.

VIII. ACKNOWLEDGMENTS

The presented work was supported by NIRWARN (Near-Infrared Warning), BMBF 01M3157B, Germany.

REFERENCES

[1] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A Tutorial on Particle Filters for On-line Non-linear/Non- Gaussian Bayesian Tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, 2 2002.

[2] Y. Boers and J. Driessen. Particle filter based detection for tracking. In *Proceedings of the American Control Conference*, volume 6, pages 4393–4397, Arlington, VA, USA, 6 2001.

[3] J. Giebel, D. M. Gavrila, and S. Christoph. A Bayesian Framework for Multi-cue 3D Object Tracking. In *Computer Vision - ECCV 2004, 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part IV*, volume 3024, pages 241–252. Springer, 2004.

[4] C. Idler, R. Schweiger, D. Paulus, M. Mählisch, and W. Ritter. Realtime vision based multi-target-tracking with particle filters in automotive applications. In *IV2006, IEEE Intelligent Vehicles Symposium*, pages 188–193, Tokyo, 2006.

[5] M. Isard and A. Blake. CONDENSATION - conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.

[6] K. Okuma, A. Taleghani, N. Freitas, J. J. Little, and L. D. G. A Boosted Particle Filter: Multitarget Detection and Tracking. In *Computer Vision - ECCV 2004, 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part I*, volume 3021, pages 28–39. Springer, 2004.

[7] C. Papageorgiou, M. Oren, and T. Poggio. A general framework for object detection. In *Proceedings of the Sixth International Conference on Computer Vision (ICCV'98)*, pages 555–562, Bombay, India, 1 1998.

[8] V. Philomin, R. Duraiswami, and L. Davis. Pedestrian tracking from a moving vehicle. In *Proceedings of the IEEE Intelligent Vehicles Symposium, 2000*, pages 350–355, Dearborn, MI, USA, 10 2000.

[9] W. Press, B. P. Flannery, S. Teukolsky, and W. Vetterling. *Numerical Recipes in C*. Cambridge University Press, 1993.

[10] A. Shashua, Y. Gdalyahu, and G. Hayun. Pedestrian detection for driving assistance systems: Single-frame classification and system level performance. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV2004)*, pages 1–6, Parma, Italy, 6 2004.

[11] L. D. Stone, T. L. Corwin, and C. A. Barlow. *Bayesian Multiple Target Tracking*. Artech House Publishers, 10 1999.

[12] P. Viola and M. Jones. Rapid object detection using a boosted cascade of. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 511–518, Kauai, Hawaii, 12 2001. IEEE Computer Society.

[13] F. Xu and K. Fujimura. Pedestrian Detection and Tracking With Night Vision. *IEEE Transactions on Intelligent Transportation Systems*, pages 63–71, 2005.