Author manuscript, published in Pattern Recognition Letters, Special Issue on Depth Image Analysis, vol. 50, pp. 149–158, 2014

3D People Surveillance on Range Data Sequences of a Rotating Lidar

Csaba Benedek

Distributed Events Analysis Research Laboratory, Institute for Computer Science and Control, Hungarian Academy of Sciences
H-1111, Kende u. 13-17, Budapest, Hungary, e-mail: benedek.csaba@sztaki.mta.hu

Abstract

In this paper, we propose an approach on real-time 3D people surveillance, with probabilistic foreground modeling, multiple person tracking and on-line re-identification. Our principal aim is to demonstrate the capabilities of a special range sensor, called rotating multi-beam (RMB) Lidar, as a future possible surveillance camera. We present methodological contributions in two key issues. First, we introduce a hybrid 2D–3D method for robust foreground-background classification of the recorded RMB-Lidar point clouds, with eliminating spurious effects resulted by quantification error of the discretized view angle, non-linear position corrections of sensor calibration, and background flickering, in particularly due to motion of vegetation. Second, we propose a real-time method for moving pedestrian detection and tracking in RMB-Lidar sequences of dense surveillance scenarios, with short- and long-term object assignment. We introduce a novel person re-identification algorithm based on solely the Lidar measurements, utilizing in parallel the range and the intensity channels of the sensor, which provide biometric features. Quantitative evaluation is performed on seven outdoor Lidar sequences containing various multi-target scenarios displaying challenging outdoor conditions with low point density and multiple occlusions.

Key words: rotating multi-beam Lidar, MRF, motion segmentation, re-identification

1. Introduction

Moving people detection, localization and tracking are important issues in intelligent surveillance applications, such as person counting, activity recognition or abnormal event detection. However, these tasks are still challenging in crowded outdoor scenes due to uncontrolled illumination conditions, irrelevant background motion, and occlusions caused by various moving and static scene objects.

Vision algorithms in surveillance systems often follow a sequential approach (Mitzel et al., 2010), starting from low level classification of the observed environment, until object level and event level analysis of the scene. Foreground segmentation is a crucial initial step (Benedek et al., 2012), since apart from highlighting the regions of interest, accurate object-silhouette masks can directly provide useful information for the scene interpretation modules, like biometric descriptors or various indicators of human behavior. Errors in the extracted foreground mask may also effect the consecutive person localization (Utasi and Benedek, 2011) and tracking (Baltieri et al., 2011) steps, especially in scenes with strong vegetation motion and occlusion. Model-based person tracking algorithms are widely used in the literature. An approach on 3D estimation of human pose from a monocular video was proposed by (Brubaker et al., 2010), which adopts a physics-based model. In (Plaenkers and Fua, 2002), a model-based technique has been introduced to extract the silhouettes of moving people from stereo video sequences, and synthesizing realistic 3D person models. In both cases, however, a single person can be observed in each video frame, which condition is often not valid for outdoor surveillance scenes. (Shu et al., 2012) introduced a part-based human detector, which builds on person-specific SVM classifiers capturing the articulations of the human bodies in dynamically changing appearance and background. For such black-box models, an extensive training set selection is a crucial step.

Person re-identification is a fundamental task both for connecting the erroneously broken trajectories of the short term tacker module, and for identifying people who temporarily leave the Field of View (FoV) and re-appear later. Numerous methods in the literature address person re-identification in optical videos (Bak et al., 2010; Farenzena et al., 2010; Prosser et al., 2010), however, their objectives are often notably different from the needs in our focused application. In the referred works, people identification is fulfilled within a large database (>100 people) using a ranking system, and the applied evaluation metric favors already, if the correct match is included within the first few candidates. This condition is acceptable if a manual verification follows the automated identification step (e.g. search in a police database), but in a fully automated surveillance system each person should be labeled with a single unambiguous identifier in real-time. On the other hand, we only deal with a few (6-8) pedestrians within a scenario, which enables us to use weak biometric features for identification. Previously, (Baltieri et al., 2011) introduced a complete 3D video surveillance system implementing model based person tracking with re-identification based on multiple camera inputs, however it uses a computationally expensive Marked Point Process based approach for the localization, which currently does not enable real-time performance. Another practical problem is that multiple camera systems should usually be carefully fixed and calibrated beforehand, which makes quick temporary installation difficult for applications monitoring customized events.

Range image sequences offer significant advantages versus conventional video flows for scene analysis, since geometrical information is directly available (Schiller and Koch, 2011), which can provide more reliable features than intensity, color or texture values (Wang et al., 2006; Benedek and Szirányi, 2008). Using Time-of-Light (ToF) cameras (Schiller and Koch, 2011) or scanning Lidar sensors (Kaestner et al., 2010) enable recording range images independently of the illumination conditions and we can also avoid artifacts of stereo vision techniques. From the point of view of data analysis, ToF cameras record depth image sequences over a regular 2D pixel lattice, where established image processing approaches, such as Markov Random Fields (MRFs) can be adopted for smooth and observation consistent segmentation and recognition (Benedek and Szirányi, 2008). However, such cameras have a limited Field of View (FoV), which can be a drawback for surveillance and monitoring applications.

Rotating multi-beam Lidar systems (RMB-Lidar) provide a 360° FoV of the scene, with a vertical resolution equal to the number of the sensors, while the horizontal angle resolution depends on the speed of rotation (see Fig. 1). Each laser point of the output point cloud is associated with 3D spatial coordinates and a calibrated intensity value of the laser reflection which is related to the material and surface properties of the target point. For efficient data processing, the 3D RMB-Lidar points are often projected onto a cylinder shaped range image (Kaestner et al., 2010; Kalyan et al., 2010). However, this mapping is usually ambiguous: On one hand, several laser beams with slight orientation differences are assigned to the same pixel, although they may return from different surfaces. As a consequence, a given pixel of the range image may represent different background objects at the consecutive time steps. This ambiguity can be moderately handled by applying multi-modal distributions in each pixel for the observed background-range values (Kaestner et al., 2010), but the errors quickly aggregate in case of dense background motion, which can be caused e.g. by moving vegetation. On the other hand, due to physical considerations, the raw data of distance, pitch and angle provided by the RMB-Lidar sensor must undergo a strongly nonlinear calibration step to obtain the Euclidean point coordinates (Muhammad and Lacroix, 2010), therefore, the density of the points mapped to the regular lattice of the cylinder surface may be inhomogeneous. To avoid the above artifacts of background modeling, (Kalyan et al., 2010) has directly extracted the foreground objects from the range image by mean-shift segmentation and blob detection. However, we have experienced that if the scene has simultaneously several moving and static objects in a wide distance range, the moving pedestrians are often merged into the same blob with neighboring scene elements.

Instead of projecting the points to a range image, another way is to interpret the scene in the spatial 3D domain. MRF-like techniques based on 3D spatial point neighborhoods are frequently applied in remote sensing for point cloud classification (Lafarge and Mallet, 2012), however the accuracy is low in case of small neighborhoods, otherwise the computational complexity rapidly increases. In (Spinello et al., 2010, 2011) methods have been introduced for 3D pedestrian detection and tracking in point cloud streams of a mobile RMB-Lidar sensor, where the main challenge was to distinguish the pedestrians from other street objects within a large FoV with compensating the sensor motion. In this paper, we address significantly different scenarios: we use the RMB-Lidar sensor in a fixed position, and monitor a dense scene with several moving people in a compact outdoor environment, such as a courtyard or a small square. We expect high occlusion rate between the observed people due to crossing trajectories, and the considered pedestrians may leave the FoV and re-appear at any time during the inspection.

The main contributions of our method are twofold. *Firstly*, we introduce a hybrid 2D–3D approach (partially presented in Benedek et al. (2012)) for dense foreground-background segmentation of RMB-Lidar point cloud sequences obtained from a fixed sensor position. Our technique solves the computationally critical spatial filtering steps in the 2D range image domain by an MRF model, however, ambiguities of discretization are handled by joint consideration of true 3D positions and back projection of 2D labels. By developing a spatial foreground model, we significantly decrease the spurious effects of irrelevant background motion, which principally caused by moving tree crowns and bushes. For quantitative point level evaluation, we have developed a 3D point cloud Ground Truth (GT) annotation tool, and compared the detection results of the proposed model to three reference methods.

Secondly, we propose a real-time method for moving pedestrian detection and tracking in RMB-Lidar sequences for dense surveillance scenarios, with short- and long-term object assignment. Our tracker is non-model-based, using the assumption that people movements are expected in the monitored scene. During the Short-Term Assignment (STA) the different people are separated in the foreground regions of the point cloud frames, and the corresponding centroid positions are assigned to each other over the consecutive time frames. The Long-Term Assignment (LTA) is responsible for connecting the broken trajectories caused by STA errors and identifying the re-appearing people. This step is accomplished by extracting simple discriminative features from the tracked object sequences, and these descriptors are archived if the object disappears from the FoV. For newly appearing objects the descriptors are extracted over an initialization period, then re-activation is based on matching a given new object with its possible archived or temporarily invisible predecessors. As a consequence, in our system the STA of the tracking process can be obtained in real-time, while the identification information is displayed with a few seconds *delay* after the target had re-appeared. As a key novelty of the proposed system, the weak biometric features used for person re-identification are solely derived from the Lidar measurements, by exploiting in parallel the range and the intensity channels of the sensor. We propose here a combination of descriptors featuring the clothing and the height of the tracked pedestrians. The tracker module is quantitatively evaluated in seven challenging surveillance sequences, by measuring the accuracy both of STA and LTA.

An important aim of this paper is also to investigate the efficiency of the RMB-Lidar sensor as a surveillance camera. Therefore during the tests we *did not use* any additional sensors, such us optical or thermal cameras to support the tracking and re-identification steps, which purely exploit the 3D point position and intensity information of the Lidar. Although we also recorded the test scenarios with an optical camera, these videos are only used for validation of re-identification. In this way, our system does not need any additional scene specific calibration step thus it can be very quickly installed, or the current viewpoint configuration can be modified.

2. Problem formulation and data mapping

Assume that the RMB-Lidar system contains R vertically aligned sensors, and rotates around a fixed axis with a possibly varying speed[1]. The output of the Lidar within a time frame t is a *point cloud* of $l^t = R \cdot c^t$ points: $\mathcal{L}^t = \{p_1^t, \dots, p_{l^t}^t\}$. Here c^t is the number of point *columns* obtained at t, where a given column contains R concurrent measurements of the R sensors, thus c^t depends on the rotation speed. Each point, $p \in \mathcal{L}^t$, is associated to sensor distance $d(p) \in [0, D_{\max}]$, pitch index $\hat{\vartheta}(p) \in \{1, \dots, R\}$ and yaw angle $\varphi(p) \in [0, 360°]$ parameters. $d(p)$ and $\hat{\vartheta}(p)$ are directly obtained from the Lidar's data flow, by taking the measured distance and sensor index values corresponding to p. Yaw angle $\varphi(p)$ is calculated from the Euclidean coordinates of p projected to the ground plane, since the R sensors have different horizontal view angles, and the angle correction of calibration may also be significant (Muhammad and Lacroix, 2010). Apart from the geometric parameters, each point p has a calibrated intensity value, denoted by $g(p)$.

For efficient data manipulation, we also introduce a range image mapping of the obtained 3D data. We project the point cloud to a cylinder, whose central basis point is the ground position of the RMB-Lidar and the axis is prependicular to the ground plane. Note that slightly differently from (Kalyan et al., 2010), this mapping is also efficiently suited to configurations, where the Lidar axis is tilted do increase the vertical Field of View. Then we stretch a $S_H \times S_W$ sized 2D pixel lattice S on the cylinder surface, whose height S_H is equal to the R sensor number, and the width S_W determines the fineness of discretization of the yaw angle. Let us denote by s a given pixel of S, with $[y_s, x_s]$ coordinates. Finally, we define the $\mathcal{P} : \mathcal{L}^t \to S$ point mapping operator, so that y_s is equal to the pitch index of the point and x_s is set by dividing the $[0, 360°]$ domain of the yaw angle into S_W bins:

$$s \overset{\text{def}}{=} \mathcal{P}(p) \text{ iff } y_s = \hat{\vartheta}(p), \; x_s = \text{round}\left(\varphi(p) \cdot \frac{S_W}{360°}\right) \quad (1)$$

3. Foreground-background separation

The goal of the foreground detector module is at a given time frame t to assign each point $p \in \mathcal{L}^t$ to a label $\omega(p) \in \{\text{fg}, \text{bg}\}$ corresponding to the moving object (i.e. foreground, fg) or background classes (bg), respectively.

3.1. *Background model*

The background modeling step assigns a fitness term $f_{\text{bg}}(p)$ to each $p \in \mathcal{L}^t$ point of the cloud, which evaluates the hypothesis that p belongs to the background. The process starts with a cylinder mapping of the points based on (1), where we use a $R \times S_W^{\text{bg}}$ pixel lattice S^{bg} (R is the sensor number). Similarly to (Kaestner et al., 2010), for each s cell of S^{bg}, we maintain a Mixture of Gaussians (MoG) approximation of the $d(p)$ distance histogram of p points being projected to s. Following the approach of (Stauffer and Grimson, 2000), we use a fixed K number of components (here $K = 5$) with weight w_s^i, mean μ_s^i and standard deviation σ_s^i parameters, $i = 1 \dots K$. Then we sort the weights in decreasing order, and determine the minimal k_s integer which satisfies $\sum_{i=1}^{k_s} w_s^i > T_{\text{bg}}$ (we used here $T_{\text{bg}} = 0.89$). We consider the components with the k_s largest weights as the background components. Thereafter, denoting by $\eta()$ a Gaussian density function, and by $\mathcal{P}^{\text{bg}}$ the projection transform onto S^{bg}, the $f_{\text{bg}}(p)$ background evidence term is obtained as:

$$f_{\text{bg}}(p) = \sum_{i=1}^{k_s} w_s^i \cdot \eta\left(d(p), \mu_s^i, \sigma_s^i\right), \text{ where } s = \mathcal{P}^{\text{bg}}(p). \quad (2)$$

The Gaussian mixture parameters are set and updated based on (Stauffer and Grimson, 2000), while we used $S_W^{\text{bg}} = 2000$ angle resolution, which provided the most efficient detection rates in our experiments. By thresholding $f_{\text{bg}}(p)$, we can get a dense foreground/background labeling of the point cloud (Kaestner et al., 2010; Stauffer and Grimson, 2000) (referred later as *Basic MoG* method), but as shown in the first row of Fig. 8, this classification is notably noisy in scenarios recorded in large outdoor scenes.

3.2. *DMRF approach on foreground segmentation*

In this section, we propose a Dynamic Markov Random Field (DMRF) model to obtain smooth, noiseless and observation consistent segmentation of the point cloud sequence. Since MRF optimization is computationally intensive (Boykov and Kolmogorov, 2004), we define the DMRF model in the range image space, and 2D image segmentation is followed by a point

[1] The speed of rotation can often be controlled by software, but even in case of constant control signal, we must expect minor fluctuations in the measured angle-velocity, which may result in different number of points for different 360° scans in time.

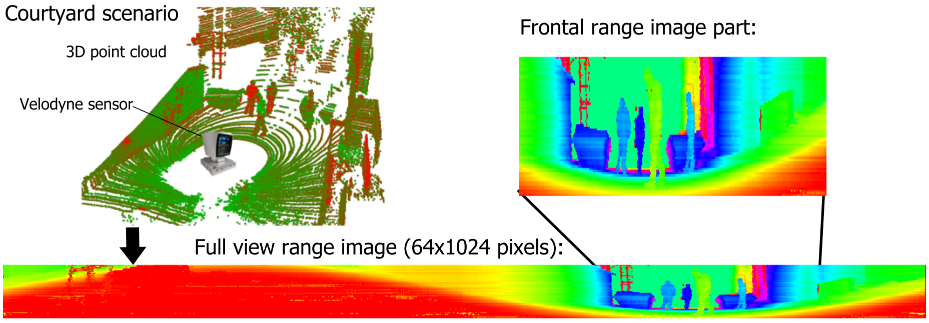
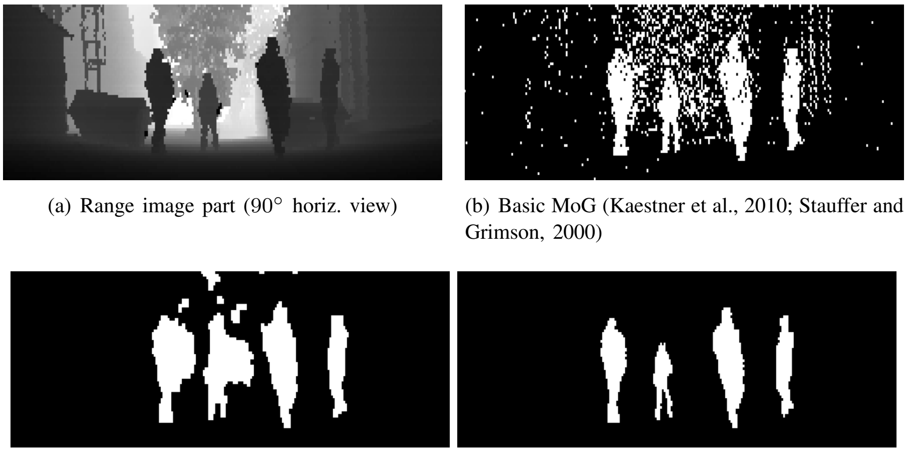

Courtyard scenario
3D point cloud
Velodyne sensor
Frontal range image part:
Full view range image (64x1024 pixels):

Fig. 1. Point cloud recording and range image formation with a Velodyne HDL-64E RMB-Lidar sensor



(a) Range image part (90° horiz. view)

(b) Basic MoG (Kaestner et al., 2010; Stauffer and Grimson, 2000)

(c) uniMRF (Wang et al., 2006)

(d) Proposed DMRF segmentation

Fig. 2. Foreground segmentation in a range image part with three different methods

classification step to handle ambiguities of the mapping. As defined by (1) in Sec. 2, we use a $\mathcal{P}$ cylinder projection transform to obtain the range image, with a $S_W = \hat{c} < S_W^{\text{bg}}$ grid with, where $\hat{c}$ denotes the expected number of point columns of the point sequence in a time frame. By assuming that the rotation speed is slightly fluctuating, this selected resolution provides a dense range image, where the average number of points projected to a given pixel is around 1. Let us denote by $P_s \subset \mathcal{L}^t$ the set of points projected to pixel s. For a given direction, foreground points are expected being closer to the sensor than the estimated mean background range value. Thus, for each pixel s we select the closest projected point $p_s^t = \text{argmin}_{p \in P_s} d(p)$, and assign to pixel s of the range image the $d_s^t = d(p_s^t)$ distance value. For 'undefined' pixels ($P_s = \emptyset$), we interpolate the distance from the neighborhood. For spatial filtering, we use an eight-neighborhood system in S, and denote by $N_s \subset S$ the neighbors of pixel s.

Next, we assign to each $s \in S$ foreground and background energy (i.e. negative fitness) terms, which describe the class memberships based on the observed $d(s)$ values. The background energies are directly derived from the parametric MoG probabilities using (2):

$$\varepsilon_{\text{bg}}^t(s) = -\log\left(f_{\text{bg}}(p_s^t)\right).$$

For description of the foreground, using a constant ε_{fg} could be a straightforward choice (Wang et al., 2006) (we call this approach *uniMRF*), but this uniform model results in several false alarms due to background motion and quantization artifacts. Instead of temporal statistics, we use spatial distance similarity information to overcome this problem by using the following assumption: whenever s is a foreground pixel, we should find foreground pixels with similar range values in the neighborhood (Fig. 3 top). For this reason, we use a non-parametric kernel density model for the foreground class:

$$\varepsilon_{\text{fg}}^t(s) = \sum_{r \in N_s} \zeta(\varepsilon_{\text{bg}}^t(r), \tau_{\text{fg}}, m_\star) \cdot k\left(\frac{d_s^t - d_r^t}{h}\right),$$

where h is the kernel bandwidth and $\zeta : \mathbb{R} \to [0,1]$ is a sigmoid function (see Fig. 3):

$$\zeta(x, \tau, m) = \frac{1}{1 + \exp(-m \cdot (x - \tau))}.$$

We use here a uniform kernel: $k(x) = \mathbf{1}\{|x| \leq 1\}$, where $\mathbf{1}\{.\} \in \{0,1\}$ is the binary indicator function of a given event.

To formally define the range image segmentation task, to each pixel $s \in S$, we assign a $\omega_s^t \in \{\text{fg}, \text{bg}\}$ class label so that we aim to minimize the following energy function:

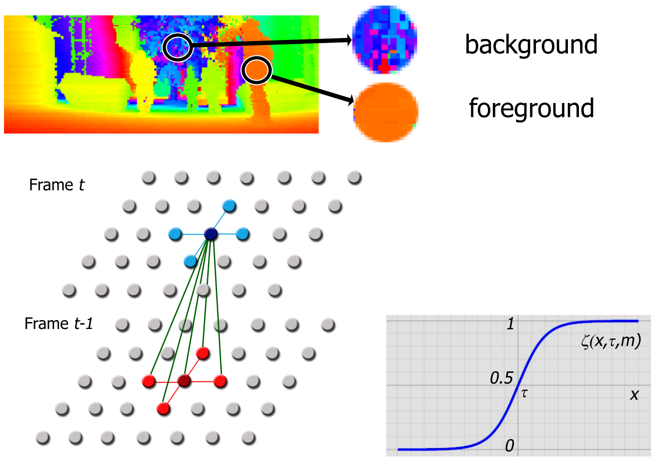


Fig. 3. Top: demonstrating the different local range value distributions in the neighborhood of a given foreground and background pixel, respectively. Bottom: structure of the dynamic MRF model, and plot of the used sigmoid function

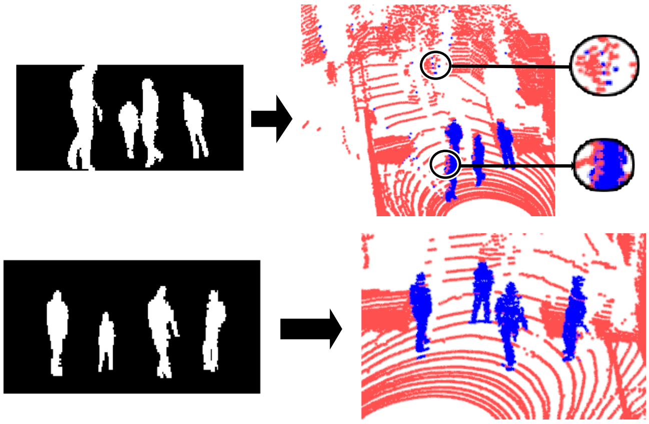


Fig. 4. Backprojection of the range image labels to the point cloud. Top: simple backprojection with assigning the same label to s and p, whenever $s = \mathcal{P}(p)$. Bottom: result of the proposed backprojection scheme

$$E = \sum_{s\in S} V_D(d_s^t|\omega_s^t) + \sum_{s\in S} \underbrace{\sum_{r\in N_s} \alpha \cdot \mathbf{1}\{\omega_s^t \neq \omega_r^{t-1}\}}_{\xi_s^t}$$

$$+ \sum_{s\in S} \underbrace{\sum_{r\in N_s} \beta \cdot \mathbf{1}\{\omega_s^t \neq \omega_r^t\}}_{\chi_s^t}, \qquad (3)$$

where $V_D(d_s^t|\omega_s^t)$ denotes the data term, while ξ_s^t and χ_s^t are the temporal and spatial smoothness terms, respectively, with $\alpha > 0$ and $\beta > 0$ constants. Let us observe, that although the model is dynamic due to dependencies between different time frames (see the ξ_s^t term), to enable real time operation, we develop a causal system, i.e. labels from the past are not updated based on labels from the future.

The data terms are derived from the data energies by sigmoid mapping:

$$V_D(d_s^t|\omega_s^t = \text{bg}) = \zeta(\varepsilon_{\text{bg}}^t(s), \tau_{\text{bg}}, m_{\text{bg}})$$

$$V_D(d_s^t|\omega_s^t = \text{fg}) = \begin{cases} 1, & \text{if } d_s^t > \max\limits_{\{i=1\dots k_s\}} \mu_s^{i,t} + \epsilon \\ \zeta(\varepsilon_{\text{fg}}^t(s), \tau_{\text{fg}}, m_{\text{fg}}), & \text{otherwise.} \end{cases}$$

The sigmoid parameters τ_{fg}, τ_{bg}, m_{fg}, m_{bg} and $m_\star$ can be estimated by Maximum Likelihood strategies based on a few manually annotated training images. As for the smoothing factors, we use $\alpha = 0.2$ and $\beta = 1.0$ (i.e. the spatial constraint is much stronger), while the kernel bandwidth is set to $h = 30$cm. The MRF energy (3) is minimized via the fast graph-cut based optimization algorithm (Boykov and Kolmogorov, 2004).

The result of the DMRF optimization is a binary foreground mask on the discrete S lattice. As shown in Fig. 4, the final step of the method is the classification of the points of the original $\mathcal{L}$ cloud, considering that the projection may be ambiguous, i.e. multiple points with different true class labels can be projected to the same pixel of the segmented range image. With denoting by $s = \mathcal{P}(p)$ for time frame t, we use the following strategy:

- $\omega(p) = \text{fg}$, iff one of the following two conditions holds:
  (a) $\omega_s^t = \text{fg}$ and $d(p) < d_s^t + 2 \cdot h$
  (b) $\omega_s^t = \text{bg}$ and $\exists r \in N_r : \{\omega_r^t = \text{fg}, |d_r^t - d(p)| < h\}$
- $\omega(p) = \text{bg}$: otherwise.

The above constraints eliminate several (a) false positive and (b) false negative foreground points, projected to pixels of the range image near the object edges, which improvement can be seen by comparing the top and bottom examples of Fig. 4.

4. Pedestrian detection and multi-target tracking

In this section, we introduce the pedestrian tracking module of the system. The input of this step is a RMB-Lidar point cloud sequence, where each point is marked with a segmentation label of foreground or background, while the output consists of clusters of foreground regions so that the points corresponding to the same person receive the same label over the sequence. We also generate a 2D trajectory of each pedestrian.

The module iterates foot point candidate detection and position assignment steps. Although, as detailed later, we should expect several false and missing alarms among the detected pedestrian positions, we can take the advantage that RMB-Lidar point cloud sequences have nowadays notably high spatial accuracy (less than 2cm error) and high frame rate (15 Hz). For these reasons, outlier positions can be efficiently filtered by temporal analysis. Trajectory initialization is implemented in a straightforward way: we consider each target candidate position in the first point cloud frame as the initial point of a possible trajectory. In the following frames, each detected position is either assigned to an existing trajectory, or it is marked as the starting point of a new track. False alarms are removed by deleting short trajectories during the process.

4.1. *Separation of moving pedestrians*

In the starting step of the module, we estimate the footprint positions of the pedestrians in each Lidar frame. First, we fit a regular rectangular lattice C onto the ground plane, where the ground position of the Lidar system is in the central cell of C, denoted by c_0. Next the foreground regions are vertically projected onto the lattice, and at each cell, $c \in C$ we count the

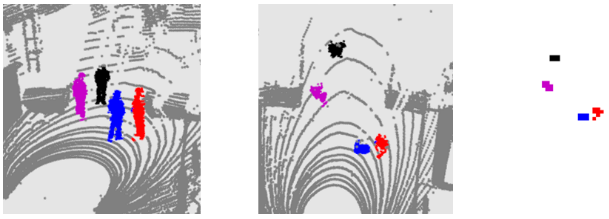


Fig. 5. Pedestrian separation. Left: side view of the segmented scene, centered: top view, right: projected blobs in the image plane

number of foreground points, $N(c)$, which are projected to c. Then a binary $N_b(.)$ cell mask is derived by thresholding $N(.)$, i.e. by selecting the cells which contain at least τ_N points. The τ_N threshold is determined so that we attempt to extract each pedestrian center from top view, but also avoid to merge closely located, or slightly connecting people (e.g. shaking the hand of each other) into the same blob in the N_b mask (used $\tau_N = 10$).

In the next step, we extract the connected components in the N_b binary image: $\{b_1, \ldots, b_k\}$, where $\forall i : b_i \subset C$. For each blob b_i we determine the "point volume" of the component as $v_i = \sum_{c \in b_i} N(c)$ and the weighted central point $c_i = \sum_{c \in b_i} c \cdot N(c)/v_i$. Considering that the point density provided by the RMB-Lidar system decreases proportionately to the squared distance from the Lidar center, we accept b_i as a valid object candidate, if $v_i \cdot ||c_i - c_0||^2 > \tau_{\text{vol}}$. We used $\tau_{\text{vol}} = 100000$ in a courtyard with a 15m radius, by measuring the point coordinates in centimeters. The output of this step is a set of the *M*easured pedestrian foot-positions in the 2D ground plane $\{M_1, \ldots, M_n\}$, where $n \leq k$ and $M_i = c_j$ if b_j is the ith valid object candidate. For visualization and later feature extraction, the foot blobs around the valid measurement points are vertically backprojected the foreground regions of the 3D point cloud, and the point cloud parts corresponding to the measurements are extracted and stored for the tracking step.

The result of the object separation step is demonstrated in Fig. 5 from different viewpoints. Note that here the tightly connecting people may be merged into the same object candidate, or blobs of partially occluded pedestrians may be missing or broken into several parts. Instead of proposing various heuristic rules to eliminate these artifacts at the level of the individual time frames, we developed a robust multi-tracking module which efficiently handles the problems at sequence level.

4.2. *Pedestrian tracking*

The pedestrian tracking module combines Short-Term Assignment (STA) and Long-Term Assignment (LTA) steps. The STA part attempts to match each actually detected object candidate (Sec. 4.1) with the current object trajectories maintained by the tracker, by purely considering the projected 2D centroid positions of the target. The STA process should also be able to continue a given trajectory if the detector misses the concerning object in a few frames due to occlusion. In these cases the temporal discontinuities of the tracks must be filled with estimated position values. On the other hand, the LTA module is responsible for extracting discriminative features for re-identification of objects lost by STA due to occlusion in many consecutive frames or leaving the FoV. For this reason, lost objects are registered to an archived object list, which is periodically checked by the LTA process. LTA should also recognize if a new person appears in the scene, who was not registered by the tracker beforehand.

4.2.1. *Short-Term Assignment (STA)*

Based on the obtained 2D object foot-positions, the Short-Term Assignment (STA) task can be formulated as a multi-target tracking problem, which is handled by a classical linear Kalman filtering approach. On each current frame the n detected target candidate points have to be assigned to m tracked object models. We assume that for each $j = 1, \ldots, m$, the tracker has already assigned a O_j predicted position to the jth maintained object track, based on the target's motion history. As introduced in Sec. 4.1, let us denote by M_i $(i = 1, \ldots, n)$ the target positions (i.e. Measurements) detected in the current frame. A distance matrix D is calculated by simple Euclidean distance in the 2D space $D_{ij} = ||M_i - O_j||$.

Based on the calculated distances, the trajectories and the current measurements are assigned with the Hungarian method (Kuhn, 1955), which expects a squared $D = [D_{ij}]_{\hat{n} \times \hat{n}}$ distance matrix, where $\hat{n} = \max\{m, n\}$. For this reason, if $m > n$ we temporarily generate $m - n$ fictional measurements which have maximum distance from all trajectories within the normalized data cube. Similarly, if $n > m$, we generate $n - m$ fictional tracks to complete the D matrix.

The output of the Hungarian matcher is a unique assignment $i \rightarrow A(i)$ between the measurements and the trajectories, where i (resp. $A(i)$) index may also correspond to a real or fictive measurement (resp. trajectory). Let τ_{dist} be a distance threshold. The obtained assignment is interpreted in the following way:

```
if $(i \leq n,\ A(i) \leq m)$:
    if $\left(D_{i,A(i)} < \tau_{\text{dist}}\right)$
        measurement $M_i$ is matched to trajectory $O_{A(i)}$
    else
        both the $i$th measurement and the $A(i)$th trajectory
        are marked as unmached.
    endif
elseif $(m \geq i > n$ and $A(i) \leq m)$
    the $A(i)$th trajectory is marked as unmached.
else
    the $i$th measurement is marked as unmached.
endif
```

If the M_i measurement is matched to the O_j trajectory point, we consider that M_i corresponds to the new position of the jth target. Since the M_i foot position is estimated as the centroid of the projected silhouette, we usually observe strong measurement noise. For this reason, we maintain a linear Kalman filter for each track, which is updated in each frame with the assigned measurements values. Tracks with label unmached are not closed immediately: they are marked as *Inactive*, in which state they can spend at most T_{SIL} time frames. *Inactive* tracks also participate in the STA process, but since they do not have actual measurements, the Kalman filter of the trajectory is up-

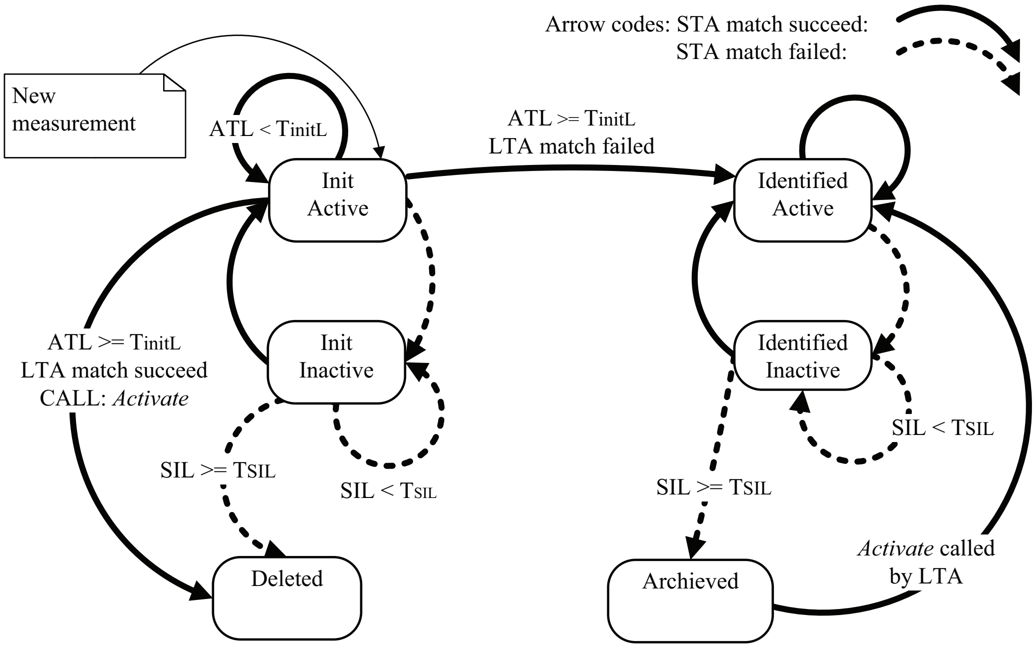


Fig. 6. State machine of the tracking algorithm. Arrows with continuous resp. dotted lines denote transition yielded by successful respectively unsuccessful Short-Term Assignment (STA) of the tracks. Further notations are as follows. ATL, Active Trajectory Length: total number of object trajectory points with valid observation values. SIL Short-term Inactivity Length: number of time frames since the object is inactive during Short-Term tracking. Tsil: maximal allowed SIL. TinitL minimal ATL for LTA-identification.

dated with the latest prediction value of the current position. In both cases, the next point of the trajectory will be the corrected state of the filter. The final step of the trajectory update is to make the Kalman prediction for the next point of each track, which can be used for measurement assignment in the next time frame. `Unmached` measurements are potential initial points of new trajectories, thus we start new object tracks for them, which is investigated during the upcoming iterations. Further management issues of `unmached` trajectories and measurements will be detailed in Sec. 4.2.3.

4.2.2. *Long-Term Assignment (LTA)*

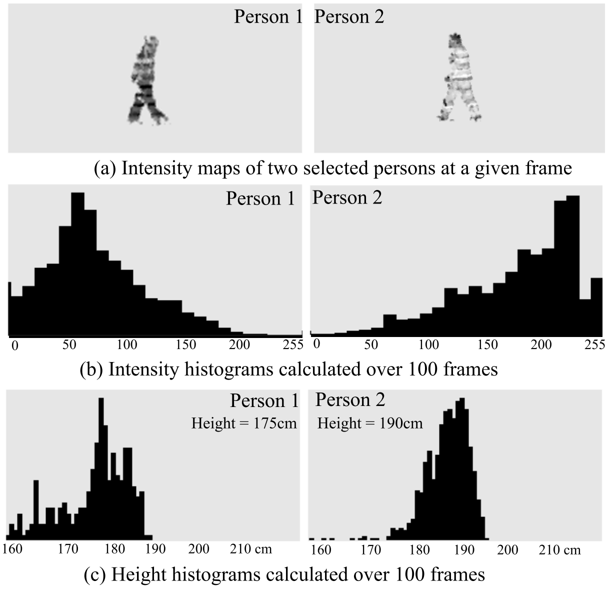
In an outdoor surveillance situation Lidar point clouds are considerably sparse. Depending on the distance from the sensor, we measured that 180-500 points correspond to a given pedestrian appearance, which encapsulate strongly limited information for biometric analysis. After investigating various static and dynamic point cloud descriptors, we found two ones as relevant for person re-identification in the considered scenes. *First*, since clothes of people consist of various materials, the calibrated reflection intensities ($g(p)$ values) obtained by the RMB-Lidar sensor exhibit different statistical characteristic for different people. Fig 7(a) displays the point silhouettes of two selected pedestrians, where points are colored by the measured laser intensity values, while Fig 7(b) shows the corresponding intensity histograms collected over 100 frames. Although the differences are usually not as significant as in this demonstration example, we found that the Bhattacharyya distance of the h_1 and h_2 *normalized* intensity histograms for two object samples efficiently indicates whether the candidates correspond to the same person or not:

$$d_{\text{Bhat}}(h_1,h_2) = -\log \sum_{k=0}^{255} \sqrt{h_1[k]\cdot h_2[k]}.$$

As a *second* feature, we measure the height of the person. In a given time frame, the height can be estimated by taking the elevation difference of the highest and lowest object points. However, this feature proved to be notably unreliable by determining it based on a single scan or only a few point clouds, due to the low vertical resolution of the RMB-Lidar camera. On the other hand we have experienced that by extracting the peak value of the *actual height histogram* over around 100 frames, we can obtain a relevant height estimation with an error less than 4cm. Even with this robust calculation, the estimated height remains a quite weak feature, but it can significantly help the long term matching process if two similarly colored people are present in the scene. Since both features are derived by temporal feature statistics, a newly appearing object must enter first an *Initial* phase, where the long-term histograms are accumulated. After a given number of frames, we can execute the LTA process which marks the object as *Identified*. We accept a long term target match only if both the intensity and the height difference features show relevant similarity. Pedestrians unsuccessfully matched to any archived objects by LTA receive a new unique identifier.

4.2.3. *Tracking process*

Based on the previously introduced STA and LTA modules, the tracking process is realized by a finite-state machine, which is displayed in Fig. 6. The state of a given actually tracked object encodes if the object is currently *Active* or *Inactive* according to the STA module, and if it is already *Identified* or is yet in the *Init*ialization phase of LTA. With these two binary pa-



(a) Intensity maps of two selected persons at a given frame

(b) Intensity histograms calculated over 100 frames

(c) Height histograms calculated over 100 frames

Fig. 7. Feature extraction for Long-Term Assignment

rameters, four states can be distinguished as shown in the top part of Fig. 6. Transitions between the corresponding *Active* and *Inactive* states are controlled by the STA module, depending on the success of matching the existing trajectories with actual measurements. *Identified* objects which are *Inactive* for more than T_{SIL} frames are moved to the archive list: *Archived* objects do not participate in the STA process, but they can be re-activated later by LTA. Objects spending T_{SIL} frames in the *Init-Inactive* state are marked as *Deleted*, and excluded from the further investigations during the tracking process. These deleted trajectories usually correspond either to measurement noise, or they are too short to provide us reliable descriptors for later LTA matching.

The LTA identification process can be applied for objects which have spent in the *Init-Active* state at least T_{initL} frames. If a match is successful with an archived object, the trajectories of the new and matched objects are merged with interpolating the missing trajectory points. Then the LTA-matched *Archived* object is moved to the *Identified-Active* state, and the new object is *Deleted* to prevent us from duplicates. On the other hand if the LTA match fails, the new object steps to the *Identified-Active* state with keeping its identifier.

4.3. *Parameter settings and practical considerations*

Since person tracking algorithms are developed for continuous operation, feasible parametrization and adaptiveness are crucial issues.

Outdoor surveillance systems using optical cameras usually suffer from external illumination changes, which can be result of either the moving position of the sun (i.e. daily illumination), or illumination changes due to changed weather circumstances (e.g. slight changes in humidity). For optical images, the above effects immediately alter the measured *color* values, thus color based appearance models of objects need usually some illumination dependent parameters, even with using illumination invariant color transforms (such as the hue channel in HSV, or a*/b* in CIE L*a*b*).

On the other hand the direct geometric information stored in the point clouds could be considered more stable, as far as the Lidar is able to operate and provide an accurate point cloud (except heavy rain or fog). From the point of view of object recognition, this feature is a great advantage compared electro-optical imaging systems, where we should train the objects or classes for differently illuminated scenarios or building up adaptive illumination following models Benedek and Szirányi (2008).

In our proposed system, the pedestrian separation and tracking modules have a few threshold-like parameters, such as the τ_N cell-occupancy value, the τ_{vol} pedestrian volume (Sec. 4.1), the τ_{dist} STA distance threshold (Sec. 4.2.1), or the T_{SIL} and T_{initL} time frame limits for *Inactive* resp. pre-*Identified* objects (Sec. 4.2.3). These factors are related either to the refreshing frequency or to the geometrical density and density-distance characteristics of the obtained point clouds, and they can be set based on the specification of the Lidar hardware. Thereafter, the thresholds can be considered constant in a scenario, with specifying the valid spatial range of the surveillance system (i.e. the field of interest).

As for intensity based person re-identification in Sec. 4.2.2, we have highly exploited that our laser scanner provides us calibrated reflectivities, thus different intensity ranges correspond to diffuse and retro-reflectors, and the observation does not significantly depend on outside illumination. In addition, laser intensity histograms are on-line re-freshed, yielding a high adaptiveness to this module. We have set the maximal allowed intensity distance for LTA matching (Sec. 4.2.2) in an empirical way, which we found it efficient for discriminating 6-8 people in several test sequences. In scenes with significantly more pedestrians it could be necessary to involve further biometric features probably from different sensors.

Another practical issue we had to deal with is related to the applied adaptive background model. According to the original background update algorithm (Stauffer and Grimson, 2000), a person standing in place for several frames becomes part of the background, and thus missed by the target detector. We handle this situation with a feedback from the object level to the low level module of the system: laser points classified as foreground points are not utilized for adaptive background update.

5. Evaluation

We have evaluated our method in 7 real outdoor Lidar sequences containing multi-target scenarios recorded in the courtyard of our institute in different parts of the year. The data flows have been captured by a Velodyne HDL-64E sensor, which operates with $R = 64$ vertically aligned beams. The sequences contain 4-8 people walking in a $220m^2$ area FoV in 1-$15m$ distances from the Lidar. The rotation speed was set from 15Hz to 20Hz. In the background, heavy motion of the vegetations make the accurate classification challenging. We have also recorded

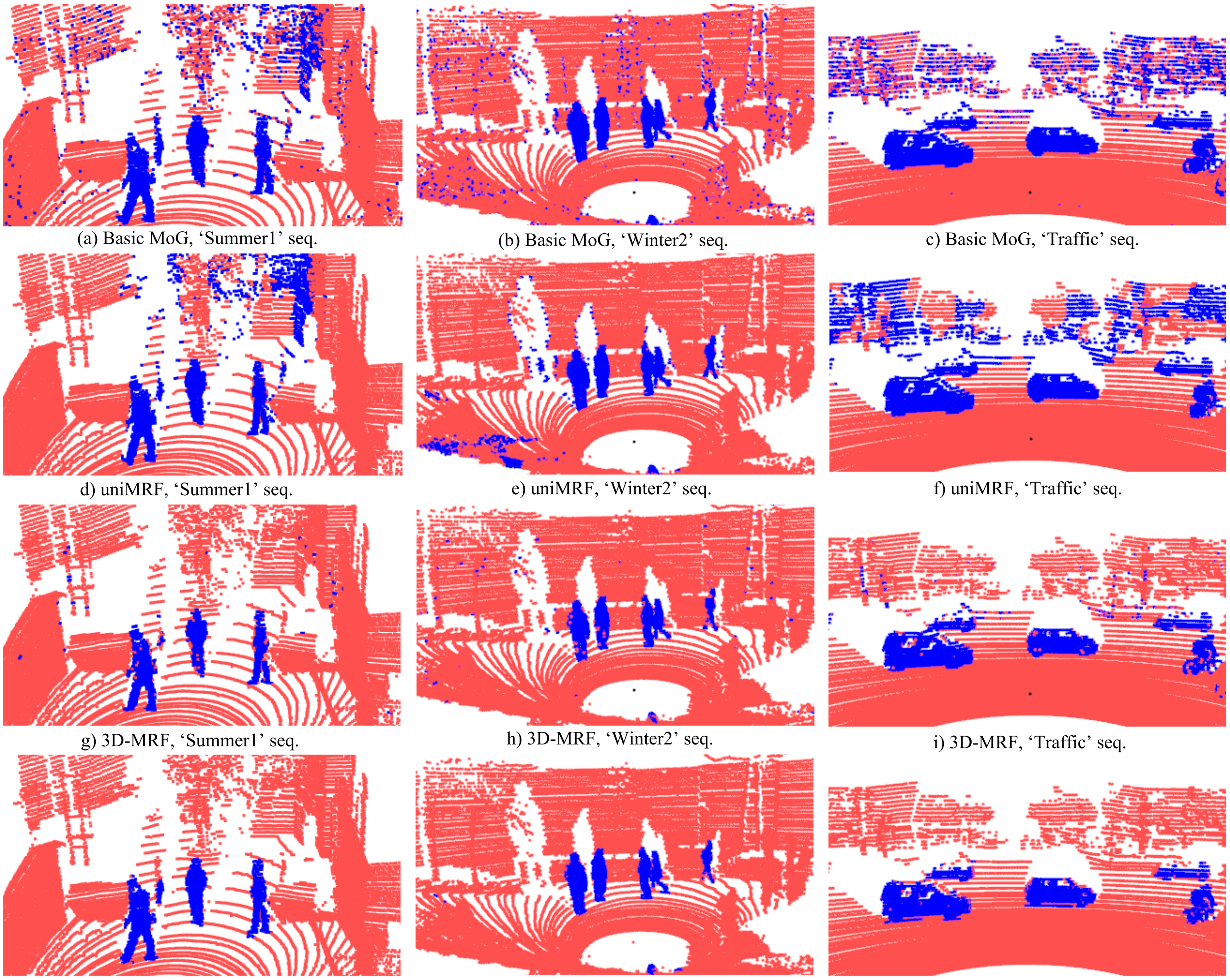

(a) Basic MoG, 'Summer1' seq.
(b) Basic MoG, 'Winter2' seq.
c) Basic MoG, 'Traffic' seq.
d) uniMRF, 'Summer1' seq.
e) uniMRF, 'Winter2' seq.
f) uniMRF, 'Traffic' seq.
g) 3D-MRF, 'Summer1' seq.
h) 3D-MRF, 'Winter2' seq.
i) 3D-MRF, 'Traffic' seq.
j) Proposed DMRF, 'Summer1' seq.
k) Proposed DMRF, 'Winter2' seq.
l) Proposed DMRF, 'Traffic' seq.

Fig. 8. Foreground classification results on sample time frames with the *Basic MoG*, *uniMRF*, *3D-MRF* and the *proposed DMRF* models: foreground points are displayed in blue (dark in gray print). First two columns correspond to people surveillance scenarios, while on the third column we can investigate the usability of the methods in a traffic monitoring environment

the test scenarios with a standard video camera *only* for verification of the tracking and re-identification process. The advantage of using sequences from different seasons was that we could test the robustness of the approach versus seasonal clothing habits (winter coats or T-Shirts) and illumination changes. Names (`Summer1`-`Spring2`) and basic properties of the test sequences are listed in Table 1.

We divided the testing phase into two parts. *First*, we have evaluated the proposed DMRF foreground-background separation process, which is a general contribution of the present work, and may be also applied in different applications from pedestrian surveillance. For this reason, as an example we also inserted a traffic monitoring (`Traffic`) scenario (see Fig. 8, third column), which sequence was recorded with 5Hz rotation speed from the top of a car waiting at a traffic light in a crowded crossroad. Here the provided point clouds are significantly larger: each scan contains around 260000 points. *Second*, we have also verified the multiple people tracking and re-identification modules by counting the correct and incorrect trajectory matches during the whole observation periods.

5.1. *Evaluation of foreground-background separation*

We have compared our proposed DMRF model for foreground-background separation to three reference solutions:

(i) *Basic MoG*, introduced in Sec. 3.1, which is based on (Kaestner et al., 2010) with using on-line K-means parameter update (Stauffer and Grimson, 2000).

(ii) *uniMRF*, introduced in Sec. 3.2, which partially adopts the uniform foreground model of (Wang et al., 2006) for range image segmentation in the DMRF framework.

(iii) *3D-MRF*, which implements a MRF model in 3D, similarly to (Lafarge and Mallet, 2012). We define here point neighborhoods in the original $\mathcal{L}^t$ clouds based on Euclidean distance, and use the background fitness values of (2) in the data model. The graph-cut algorithm (Boykov and Kolmogorov, 2004) is adopted again for MRF energy optimization.

Qualitative segmentation results on sample frames from three sequences are shown in Fig. 8, concerning the three reference methods and the proposed DMRF model. For *quantitative* (numerical) evaluation, we manually generated Ground Truth (GT). For this reason we have developed a 3D point cloud annotation tool, which enables labeling the scene regions manually as foreground or background. Next, we manually annotated around 100 relevant frames of each test sequence. For quantitative evaluation metric, we have chosen the point level F-rate of foreground detection (Benedek and Szirányi, 2008), which can be calculated as the harmonic mean of precision and recall. We have also measured the processing speed in frames per seconds (fps). The numerical performance analysis is given in Table 1(a). The results confirm that the proposed model surpasses the reference techniques in F-rate in all surveillance sequences, meanwhile the processing speed is 15-16fps, which enables real-time operation. In the `Traffic` sequence with large and dense point clouds, the 3D-MRF approach is able to slightly outperform our approach in detection rate, but the *proposed DMRF* method is significantly quicker: we measured there 2fps processing speed with 3D-MRF and 16fps with the proposed DMRF model. We can also observe that differently from 3D-MRF, our range image based technique is less influenced by the size of the point cloud.

5.2. *Evaluation of multi-target tracking*

For quantitative evaluation of the tracking process the output trajectories of the system were verified by manual observes watching the point cloud sequences and the recorded videos in parallel. (Note that the system did not use the optical video information, we only recorded it to enable verification of tracking and re-identification.)

As evaluation metrics, we counted the following events (see results in Table 1(b)):

- *STA trans. num*: number of all *Inactive→Active* state transitions during the tracking process, i.e. the number of events, when the Short-Term Assignment (STA) module can continue a track after the object had been occluded for a couple of frames (counted automatically).
- *STA trans. error*: number of erroneous track assignments by the STA module (counted manually).
- *LTA trans. num*: number of *Archived→Identified* state transitions during the tracking process, i.e. the number of events, when the Long-Term Assignment (LTA) module can recognize a previously archived and re-appearing person (counted automatically).
- *LTA trans. error*: number of erroneous person assignments by the LTA module (counted manually).

The seven surveillance sequences listed in Table 1(b) imply varying difficulty factors for the multi-target tracking process. First, we calculated the *Av*erage *people* number *per frame* (4th column) among the frames of the Lidar sequence, which contain at least two pedestrians. Higher people density results in more occlusions, thus usually in increasing *STA trans. num*, which means challenges for the STA module. On the other hand, the total number of people (4-8) and the *LTA trans. num* affect the LTA re-identification process. As shown in the table, the first three sequences have been used only to verify the STA tracking module. As for sequences `Winter1-Spring1`, by increasing the people number to 6 the re-identification step becomes crucial, but the LTA-match is still nearly faultless (97% performance). Finally, in the 8-people scenario (`Spring2`), which contains not only more people, but also a significnatly increased number of occlusions, the LTA yields 4 assignment errors out of 17 re-identification attempts, which means a 76.4% performance.

Fig. 9 displays two sample frames from the `Winter2` sequence. Between the two selected frames, all pedestrians left the FoV, therefore a complete re-assignment should have been performed by the LTA module. Note that even with applying Kalman filtering, the resulted raw object tracks are quite noisy, therefore, we applied a 80% compression of the curves in the Fourier desciptor space (Zhang and Lu, 2002), which yields the smoothed tracks displayed in Fig 9, right. A demonstration video about the tracking process in the `Winter2` sequence can be watched in the author's homepage: `http://web.eee.sztaki.hu/i4d/PRLDEMO`

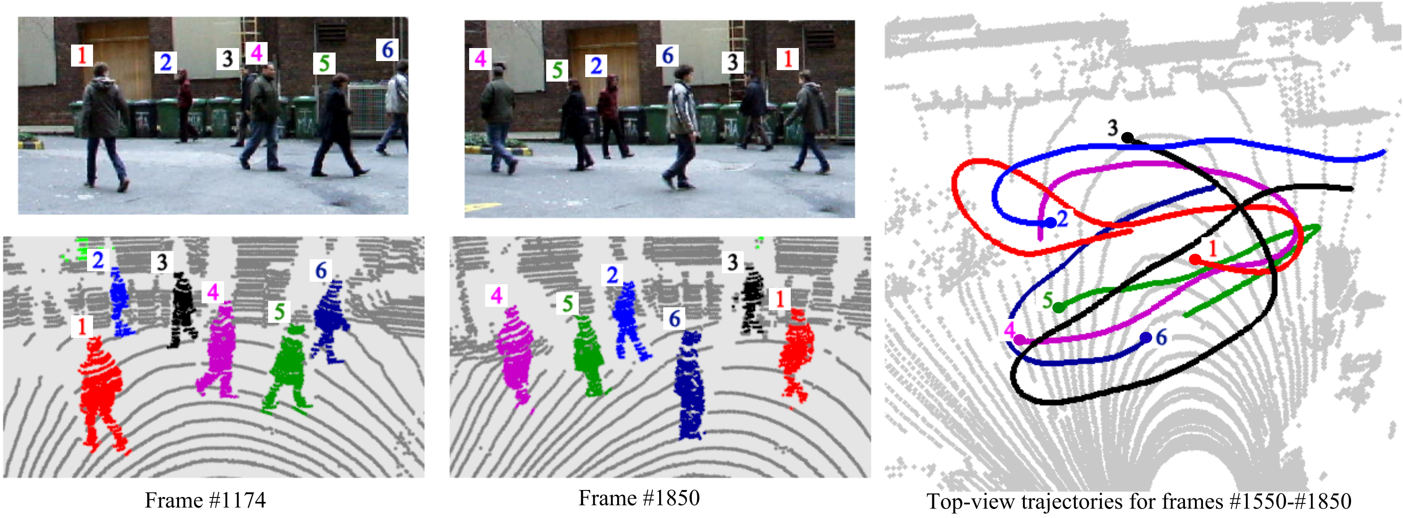
An important feature of the proposed system is the nearly real time performance with processing 15 Hz Lidar sequences. The last column of Table 1(b) lists the measured processing speed on the different test sets. Compared with fps values of Table 1(a), we can conclude that the most expensive part of the process is foreground-background segmentation (in itself 15-16 fps), since the complete workflow including foreground detection, pedestrian separation and tracking operates with 12-13 fps. We can observe a slight computational overload as the number of people increases yielding more occlusions. Quicker operation in the `Summer1` sequence is the result of the smaller point clouds, since that sequence has been recorded at 20 Hz rotation frequency.

6. Acknowledgment

This work is connected to the i4D project funded by the internal R&D grant of MTA SZTAKI, and it was also supported by the Hungarian Research Fund (OTKA #101598), and by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences. The author would like to thank his colleagues, Csaba Horváth, Dömötör Molnár and Tamás Szirányi for help in the development and implementation of the methods presented in the paper.

7. Conclusions

We have introduced a novel 3D surveillance framework for detecting and tracking multiple moving pedestrians in point clouds obtained by a rotating multi-beam (RMB) Lidar system, with focusing on specific challenges raised by the selected range sensor. We have proposed first an efficient foreground



Frame #1174 Frame #1850 Top-view trajectories for frames #1550-#1850

Fig. 9. Results of pedestrian separation and tracking in the `Winter2` Lidar sequence. Note that between the two displayed frames (#1174 and #1850) all pedestrians have left the field of interest and re-appeared in a random order, thus a complete re-identification process has been conducted. Trajectories in the right correspond to frames between #1580 and #1850, where the position in Frame #1850 is marked with a circle. Video images (in the top) were only used for validation of tracking and re-identification.

Table 1
Numerical point level evaluation of foreground detection and object level evaluation of tracking and re-identification on the test sequences

Sequence	**Point cloud**	F-measure based on 100 frames (in %)			
name	**size**	Bas. MoG	uniMRF	3D-MRF	**DMRF**
`Summer1`	65K pts/fr.	55.7	81.0	88.1	95.1
`Summer2`	86K pts/fr.	59.2	86.9	89.7	93.2
`Summer3`	86K pts/fr.	38.4	83.3	78.7	89.0
`Winter1`	86K pts/fr.	55.0	86.6	84.1	91.9
`Winter2`	86K pts/fr.	54.9	86.6	84.1	91.9
`Spring1`	86K pts/fr.	49.9	84.8	82.7	88.9
`Spring2`	86K pts/fr.	56.8	89.1	86.9	94.4
`Traffic`	260K pts/fr.	70.4	68.3	76.2	74.0
Processing Speed		120fps	17-18fps	2-7fps	15-16fps

(a) Point level evaluation of foreground detection detection accuracy (F-rate in %) and processing speed (fps, measured in a desktop computer)

Sequence	**Frame**	**People**	**Av peopl.**	**STA trans.**	**LTA trans.**	**Processing**
name	**num.**	**num.**	**per frame**	**num (error)**	**num (error)**	**speed (fps)**
`Summer1`	2556	4	3.51	57 (0)	1 (0)	14.95
`Summer2`	960	4	3.64	30 (0)	0 (-)	12.89
`Summer3`	1406	4	3.77	44 (0)	0 (-)	13.03
`Winter1`	3641	4	2.91	71 (0)	9 (0)	12.91
`Winter2`	2433	6	4.38	129 (0)	12 (0)	12.65
`Spring1`	2616	6	4.34	127 (0)	16 (1)	12.78
`Spring2`	2383	8	5.51	216 (1)	17 (4)	12.45

(b) Object level evaluation on the seven surveillance test sequences. STA: Short-Term Assignment, LTA: Long-Term Assignment. Processing speed is related to the complete workflow including foreground detection.

segmentation model, which uses a spatial foreground filter to decrease artifacts of angle quantization and background motion. This component has been quantitatively validated based on 3D Ground Truth data, and the advantages of the proposed solution versus three reference methods have been demonstrated. Thereafter, we have introduced a multi-target tracking module with on-line person re-identification functions, where biometric features were derived from the range and intensity channels of the Lidar data flow. The tracker module was also tested in real outdoor scenarios, with multiple occlusions an several re-appearing people during the observation period. The experiments confirmed, that an efficient 3D video surveillance system can be based on a single RMB-Lidar sensor, whose installation is significantly easier than setting up a calibrated multi-camera system.

References

Bak, S., Corvee, E., Bremond, F., and Thonnat, M. (2010). Person re-identification using spatial covariance regions of human body parts. In *International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 435–440.

Baltieri, D., Vezzani, R., Cucchiara, R., Utasi, Á., Benedek, C., and Szirányi, T. (2011). Multi-view people surveillance using 3D information. In *Proc. International Workshop on Visual Surveillance at ICCV*, pages 1817–1824, Barcelona, Spain.

Benedek, C., Molnár, D., and Szirányi, T. (2012). A dynamic MRF model for foreground detection on range data sequences of rotating multi-beam lidar. In *International Workshop on Depth Image Analysis, LNCS*, Tsukuba City, Japan.

Benedek, C. and Szirányi, T. (2008). Bayesian foreground and shadow detection in uncertain frame rate surveillance videos. *IEEE Transactions on Image Processing*, 17(4):608 – 621.

Boykov, Y. and Kolmogorov, V. (2004). An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137.

Brubaker, M., Fleet, D., and Hertzmann, A. (2010). Physics-based person tracking using the anthropomorphic walker. *Int. Journal of Computer Vision*, 87(1-2):140–155.

Farenzena, M., Bazzani, L., Perina, A., Murino, V., and Cristani, M. (2010). Person re-identification by symmetry-driven accumulation of local features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2360–2367.

Kaestner, R., Engelhard, N., Triebel, R., and R.Siegwart (2010). A Bayesian approach to learning 3D representations of dynamic environments. In *Proc. International Symposium on Experimental Robotics (ISER)*, Berlin. Springer Press.

Kalyan, B., Lee, K. W., Wijesoma, W. S., Moratuwage, D., and Patrikalakis, N. M. (2010). A random finite set based detection and tracking using 3D LIDAR in dynamic environments. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2288–2292, Istanbul, Turkey. IEEE.

Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval Research Logistic Quarterly*, 2:83–97.

Lafarge, F. and Mallet, C. (2012). Creating large-scale city models from 3D-point clouds: A robust approach with hybrid representation. *Int. J. of Computer Vision*.

Mitzel, D., Horbert, E., Ess, A., and Leibe, B. (2010). Multi-person tracking with sparse detection and continuous segmentation. In *European Conference on Computer Vision*, ECCV'10, pages 397–410, Berlin, Heidelberg. Springer-Verlag.

Muhammad, N. and Lacroix, S. (2010). Calibration of a rotating multi-beam Lidar. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 5648–5653, Taipei, Taiwan. IEEE.

Plaenkers, R. and Fua, P. (2002). Model-based silhouette extraction for accurate people tracking. In Heyden, A., Sparr, G., Nielsen, M., and Johansen, P., editors, *International Conference on Computer Vision*, volume 2351 of *Lecture Notes in Computer Science*, pages 325–339. Springer Berlin Heidelberg.

Prosser, B., Zheng, W.-S., Gong, S., and Xiang, T. (2010). Person re-identification by support vector ranking.

Schiller, I. and Koch, R. (2011). Improved video segmentation by adaptive combination of depth keying and Mixture-of-Gaussians. In *Proc. Scandinavian Conference on Image Analysis, Ystad, Sweden*, volume 6688 of *LNCS*, pages 59–68.

Shu, G., Dehghan, A., Oreifej, O., Hand, E., and Shah, M. (2012). Part-based multiple-person tracking with partial occlusion handling. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1815–1821.

Spinello, L., Arras, K. O., Triebel, R., and Siegwart, R. (2010). A layered approach to people detection in 3D range data. In *Proc. AAAI Conference on Artificial Intelligence*, Atlanta, Georgia, USA.

Spinello, L., Luber, M., and Arras, K. (2011). Tracking people in 3D using a bottom-up top-down detector. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1304–1310, Shanghai, China.

Stauffer, C. and Grimson, W. E. L. (2000). Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:747–757.

Utasi, Á. and Benedek, C. (2011). A 3-D marked point process model for multi-view people detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3385–3392, Colorado Springs, CO, USA.

Wang, Y., Loe, K.-F., and Wu, J.-K. (2006). A dynamic conditional random field model for foreground and shadow segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(2):279 –289.

Zhang, D. and Lu, G. (2002). A comparative study of fourier descriptors for shape representation and retrieval. In *Asian Conference on Computer Vision (ACCV*, pages 646–651. Springer.