A Dynamic MRF Model for Foreground Detection on Range Data Sequences of Rotating Multi-Beam Lidar

Csaba Benedek[1], Dömötör Molnár[12] and Tamás Szirányi[12⋆]

[1] Distributed Events Analysis Research Laboratory,
Computer and Automation Research Institute, Hungarian Academy of Sciences
Kende utca 13-17, H-1111 Budapest, Hungary
[2] Department of Information Technology, Péter Pázmány Catholic University
Práter utca 50/A, H-1083 Budapest, Hungary
`firstname.lastname@sztaki.mta.hu`

Abstract. In this paper, we propose a probabilistic approach for foreground segmentation in 360°-view-angle range data sequences, recorded by a rotating multi-beam Lidar sensor, which monitors the scene from a fixed position. To ensure real-time operation, we project the irregular point cloud obtained by the Lidar, to a cylinder surface yielding a depth image on a regular lattice, and perform the segmentation in the 2D image domain. 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, are significantly decreased by a dynamic MRF model, which describes the background and foreground classes by both spatial and temporal features. Evaluation is performed on real Lidar sequences concerning both video surveillance and traffic monitoring scenarios.

Keywords: rotating multi-beam Lidar, MRF, motion segmentation

1 Introduction

Foreground detection and segmentation are a key issues in automatic visual surveillance. Foreground areas usually contain the regions of interest, moreover, an accurate object-silhouette mask can directly provide useful information for, among others, people or vehicle detection, tracking or activity analysis.

Range image sequences offer significant advantages versus conventional video flows for scene segmentation, since geometrical information is directly available [1, 2], which can provide more reliable features than intensity, color or texture values [3, 4]. Using Time-of-Light (ToF) cameras [1] or scanning Lidar sensors

[⋆] This work was partially funded by the i4D Project of MTA SZTAKI and by the Hungarian Research Fund (Grants OTKA #76159 and #101598). C. Benedek was also supported by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences.

[5] enable recording range images independently of the outside 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 [4]. 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. For efficient data processing, the 3-D RMB-Lidar points are often projected onto a cylinder shaped range image [5, 6]. 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 [5], 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 non-linear calibration step to obtain the Euclidean point coordinates [7], 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, [6] 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 solve the foreground detection problem in the spatial 3D domain. However, 3D object level techniques principally aim to extract the bounding boxes of the pedestrians [8], instead of labeling each foreground point of the input cloud, which may be necessary for activity recognition by e.g. skeleton fitting to the silhouettes. MRF techniques based on 3D spatial point neighborhoods are frequently applied in remote sensing [9], however the accuracy is low in case of small neighborhoods, otherwise the computational complexity rapidly increases.

In this paper, we propose a hybrid approach for dense foreground-background point labeling in a point cloud obtained by a RMB-Lidar system, which monitors the scene from a fixed position. Our method 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 the true 3D positions and the 2D labels. Using a spatial foreground model, we significantly decrease the spurious effects of irrelevant background motion, which is mainly caused by moving tree crowns. We provide evaluation versus three reference methods using our new 3D point cloud Ground Truth (GT) annotation tool.

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[3]. The output of the Lidar within a time frame t is a *point cloud* of $l^t = R \cdot c^t$ points: $\mathcal{L}^t = \{p_1^t, \ldots, p_{l^t}^t\}$. 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 $\hat{\vartheta}(p) \in \{1, \ldots, R\}$ 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 [7].

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

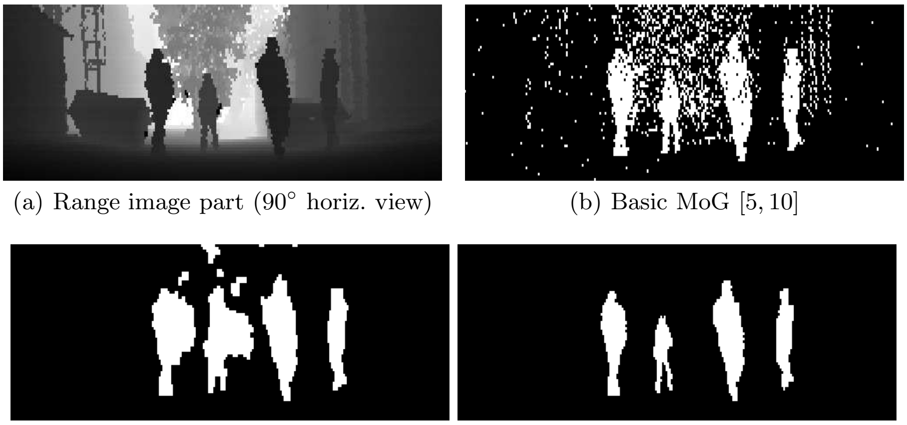
For efficient data manipulation, we also introduce a range image mapping of the obtained 3D data. We project the point cloud to a cylinder, whose central basis point is the ground position of the RMB-Lidar and the axis is prependicular to the ground plane. Note that slightly differently from [6], this 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:

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

3 Background model

The background modeling step assigns a fitness term $f_{\mathrm{bg}}(p)$ 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 $R \times S_W^{\mathrm{bg}}$ pixel lattice S^{bg} (R is the sensor number). Similarly to [5], 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 [10], 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, $i = 1 \ldots K$. Then we

[3] 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.



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

(b) Basic MoG [5, 10]

(c) uniMRF [3]

(d) Proposed DMRF segmentation

Fig. 1. Foreground segmentation in a range image part with three different methods

sort the weights in decreasing order, and determine the minimal k_s integer which satisfies $\sum_{i=1}^{k_s} w_s^i > T_{\mathrm{bg}}$(we used here $T_{\mathrm{bg}} = 0.89$). We consider the components with the k_s largest weights as the background components. Thereafter, denoting by $\eta()$ a Gaussian density function, and by $\mathcal{P}^{\mathrm{bg}}$ the projection transform onto S^{bg}, the $f_{\mathrm{bg}}(p)$ background evidence term is obtained as:

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

The Gaussian mixture parameters are set and updated based on [10], while we used $S_W^{\mathrm{bg}} = 2000$ angle resolution, which provided the most efficient detection rates in our experiments. By thresholding $f_{\mathrm{bg}}(p)$, we can get a dense foreground/background labeling of the point cloud [5, 10] (referred later as *Basic MoG* method), but as shown in Fig. 2(a),(c), this classification is notably noisy in scenarios recorded in large outdoor scenes.

4 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 [11], we define the DMRF model in the range image space, and 2D image segmentation is followed by a point 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 $S_W = \min(\hat{c}, S_W^{\mathrm{bg}}/2)$ 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. 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 $p_s^t = \operatorname{argmin}_{p \in P_s} d(p)$, and assign to pixel s of the range image the $d_s^t = d(p_s^t)$ distance value. For pixels with undefined range values ($P_s = \emptyset$), we interpolate the d_s^t 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):

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

For description of the foreground, using a constant $\varepsilon_{\mathrm{fg}}$ could be a straightforward choice [3] (we call this approach *uniMRF*), but this uniform model results in several false alarms due to background motion and quantitization 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. For this reason, we use a non-parametric kernel density model for the foreground class:

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

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

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

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 $\omega_s^t \in \{\mathrm{fg}, \mathrm{bg}\}$ class label so that we aim to minimize the following energy function:

$$E = \sum_{s \in S} V_D(d_s^t|\omega_s^t) + \sum_{s \in S} \underbrace{\sum_{r \in N_s} \alpha \cdot \mathbf{1}\{\omega_s^t \neq \omega_r^{t-1}\}}_{\xi_s^t} + \sum_{s \in S} \underbrace{\sum_{r \in N_s} \beta \cdot \mathbf{1}\{\omega_s^t \neq \omega_r^t\}}_{\chi_s^t}, \quad (3)$$

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:

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

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

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 [11].

The result of the DMRF optimization is a binary foreground mask on the discrete S lattice. 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:

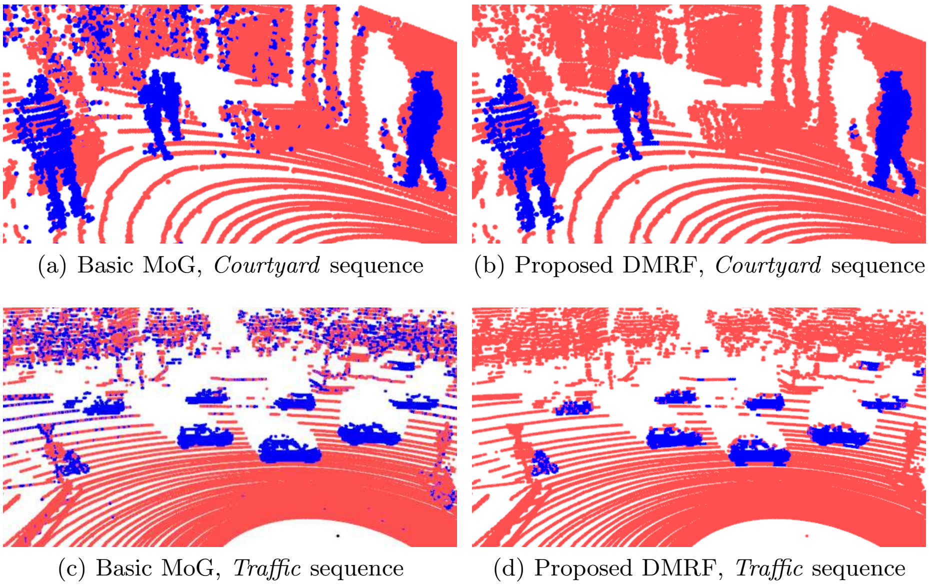
- $\omega(p) = \text{fg}$, iff one of the following two conditions holds:
  - ○ $\omega_s^t = \text{fg}$ and $d(p) < d_s^t + 2 \cdot h$ (a)
  - ○ $\omega_s^t = \text{bg}$ and $\exists r \in N_r : \{\omega_r^t = \text{fg}, |d_r^t - d(p)| < h\}$ (b)
- $\omega(p) = \text{bg}$: otherwise.

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.

5 Evaluation

We have tested our method in real Lidar sequences concerning both video surveillance (*Courtyard*) and traffic monitoring (*Traffic*) scenarios (see Fig. 2). The data flows have been recorded by a Velodyne HDL 64E S2 camera, which operates with $R = 64$ vertically aligned beams. The *Courtyard* sequence contains 2500 frames with four people walking in a $25m^2$ area in 1-5m distances from the Lidar, with crossing trajectories. The rotation speed was set to 20Hz. In the background, heavy motion of the vegetations make the accurate classification challenging. The *Traffic* sequence was recorded with 5Hz from the top of a car waiting at a traffic light in a crowded crossroad. The adaptive background model was automatically built up within a few seconds, then 160 time frames were available for traffic flow analysis. We have compared our DMRF model to three reference solutions:

1. *Basic MoG*, introduced in Sec. 3, which is based on [5] with using on-line K-means parameter update [10].
2. *uniMRF*, introduced in Sec. 4, which partially adopts the uniform foreground model of [3] for range image segmentation in the DMRF framework.
3. *3D-MRF*, which implements a MRF model in 3D, similarly to [9]. 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 [11] is adopted again for MRF energy optimization.



(a) Basic MoG, *Courtyard* sequence (b) Proposed DMRF, *Courtyard* sequence

(c) Basic MoG, *Traffic* sequence (d) Proposed DMRF, *Traffic* sequence

Fig. 2. Point cloud classification result on sample frames with the *Basic MoG* and the proposed DMRF model: foreground points are displayed in blue (dark in gray print).

Qualitative results on two sample frames are shown in Fig. 2. For Ground Truth (GT) generation, we have developed a 3D point cloud annotation tool, which enables labeling the scene regions manually as foreground or background. Next, we manually annotated 700 relevant frames of the *Courtyard* and 50 frames of the *Traffic* sequence. For quantitative evaluation metric, we have chosen the point level F-rate of foreground detection [4], 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. The results confirm that the proposed model surpasses the *Basic MoG* and *uniMRF* techniques in F-rate for both scenes, and the differences are especially notable at the *Courtyard*. Compared to the *3D-MRF* method, our model provides similar detection accuracy, but the *proposed DMRF* method is significantly quicker. Observe that differently from 3D-MRF, our range image based technique is less influenced by the size of the point cloud. In the *Traffic* sequence, which contains around 260000 points within a time frame, we measured 2fps processing speed with 3D-MRF and 16fps with the proposed DMRF model.

6 Conclusions

We have proposed a Dynamic MRF model for foreground segmentation in point clouds obtained by a rotating multi-beam Lidar system. We have introduced an efficient spatial foreground filter to decrease artifacts of angle quantitization and background motion. The model has been quantitatively validated based on

Aspect	Sequence	Seq. property	Basic MoG	uniMRF	3D-MRF	**DMRF**
Detection rate (F-rate in %)	*Courtyard*	4 obj/frame	55.7	81.0	88.1	95.1
	Traffic	20 obj/frame	70.4	68.3	76.2	74.0
Processing speed (frames per sec)	*Courtyard*	65K pts/frame	120 fps	18 fps	7 fps	16 fps
	Traffic	260K pts/frame	120 fps	18 fps	2 fps	16 fps

Table 1. Numerical evaluation on the *Courtyard* and *Traffic* sequences: detection accuracy (F-rate in %) and processing speed (fps, measured in a desktop computer)

Ground Truth data, and the advantages of the proposed solution versus three reference methods have been demonstrated. The authors thank Miklós Homolya for help in MRF code integration [11].

References

1. Schiller, I., Koch, R.: 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. (2011) 59–68
2. Langmann, B., Ghobadi, S., Hartmann, K., Loffeld, O.: Multi-modal background subtraction using gaussian mixture models. In: ISPRS Symposium on Photogrammetric Computer Vision and Image Analysis. (2010) 61–66
3. Wang, Y., Loe, K.F., Wu, J.K.: A dynamic conditional random field model for foreground and shadow segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **28**(2) (2006) 279 –289
4. Benedek, C., Szirányi, T.: Bayesian foreground and shadow detection in uncertain frame rate surveillance videos. IEEE Transactions on Image Processing **17**(4) (2008) 608 – 621
5. Kaestner, R., Engelhard, N., Triebel, R., R.Siegwart: A Bayesian approach to learning 3D representations of dynamic environments. In: Proc. International Symposium on Experimental Robotics (ISER), Berlin, Springer Press (2010)
6. Kalyan, B., Lee, K.W., Wijesoma, W.S., Moratuwage, D., Patrikalakis, N.M.: A random finite set based detection and tracking using 3D LIDAR in dynamic environments. In: IEEE International Conference on Systems, Man, and Cybernetics (SMC), Istanbul, Turkey, IEEE (2010) 2288–2292
7. Muhammad, N., Lacroix, S.: Calibration of a rotating multi-beam Lidar. In: International Conference on Intelligent Robots and Systems (IROS), Taipei, Taiwan, IEEE (2010) 5648–5653
8. Spinello, L., Luber, M., Arras, K.: Tracking people in 3D using a bottom-up top-down detector. In: IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China (2011) 1304–1310
9. Lafarge, F., Mallet, C.: Creating large-scale city models from 3D-point clouds: A robust approach with hybrid representation. Int. J. of Computer Vision (2012)
10. Stauffer, C., Grimson, W.E.L.: Learning patterns of activity using real-time tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000) 747–757
11. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Transactions on Pattern Analysis and Machine Intelligence **26**(9) (2004) 1124–1137