Author manuscript, published in International Conference on Pattern Recognition (ICPR), pp. 1379-1382, Tsukuba, Japan, 2012

Urban Traffic Monitoring from Aerial LIDAR Data with a Two-Level Marked Point Process Model

Attila Börcs and Csaba Benedek
Distributed Events Analysis Research Laboratory, Computer and Automation Research Institute H-1111, Budapest, Kende utca 13-17, Hungary, {borcs,bcsaba}@sztaki.hu

Abstract

In this paper we present a new model for joint extraction of vehicles and coherent vehicle groups in airborne LIDAR point clouds collected from crowded urban areas. Firstly, the 3D point set is segmented into terrain, vehicle, roof, vegetation and clutter classes. Then the points with the corresponding class labels and intensity values are projected to the ground plane, where the optimal vehicle and traffic segment configuration is described by a Two-Level Marked Point Process (L^2MPP) model of 2D rectangles. Finally, a stochastic algorithm is utilized to find the optimal configuration.

1. Introduction

Automatic traffic monitoring is a central goal of urban traffic control, environmental protection and aerial surveillance applications. Complex traffic analysis needs a hierarchical modeling approach: at low level *individual vehicles* should be detected and separated, meanwhile at a higher level we need to extract *coherent traffic segments*, by identifying groups of corresponding vehicles, such as cars in a parking lot, or a vehicle queue waiting in front of a traffic light. In this paper, we introduce a joint probabilistic model for vehicle detection and traffic segmentation in airborne LIDAR data, which contains point position, intensity and echo information.

LIDAR based vehicle detection methods in the literature follow generally either a grid-cell- or a 3D point-cloud-analysis-based approach [6]. We propose a hybrid model here, where the initial point cloud is classified via 3D features, but the optimal object configuration is extracted in a 2D lattice, after ground plane projection. We model a traffic scene by a Marked Point Process (MPP) [3], which is an efficient Bayesian tool to characterize object populations, through jointly describing individual objects by various data terms, and using information from entity interactions by prior geometric constraints. However, conventional MPP models offer limited options for hierarchical scene modeling, since they usually exploit pairwise object interactions, which are defined on fixed symmetric object neighborhoods. In a traffic situation we often find several groups of regularly aligned vehicles, but we must also deal with junctions or skewed parking places next to the roads (Fig. 3), where many differently oriented cars appear close to each other. In addition, the coherent car groups may have thin, elongated shapes, therefore concentric neighborhoods are less efficient. For this reason, we propose here a Two-Level MPP (L^2MPP) model, which partitionates the complete vehicle population into vehicle groups, called *traffic segments*, and extracts the vehicles and the optimal segments simultaneously by a joint energy minimization process. Object interactions are differently defined within the same segment and between two different segments, implementing adaptive object neighborhoods. This model extends our single level MPP method [2] proposed for vehicle detection. In addition, we present here an improved point cloud segmentation algorithm, and provide a detailed quantitative evaluation on four datasets of 471 vehicles, considering two reference methods [4, 5].

2. Point Cloud Preprocessing

We have developed a Markov Random Field (MRF) model for point cloud segmentation, which utilizes various 3D descriptors. For featuring the *terrain* class, we estimate the dominant plane of the input cloud using the RANSAC algorithm, and calculate the distance of each point from this plain. Regarding the *roof* class, we assume that roof points form large connected regions of the cloud, which are composed of segments with uniform surface normals. Local point cloud density is also calculated to recognize sparse *clutter* regions (like most

[1]This work was supported by the Hungarian Research Fund (OTKA #101598), and by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences.

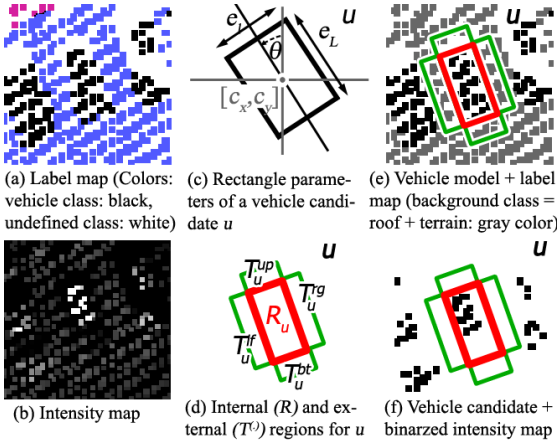

Figure 1. Data model elements

reflections from walls), and echo number is exploited for identifying vegetation. Points of vehicles appear as outliers from the previous classes, and in addition, they fit prior height distributions of the possible cars. Finally, we construct an MRF energy model based on the previous features. For optimization, the fast ICM algorithm proved to be efficient in the used test sets. After the 3D segmentation process, we stretch a 2D pixel lattice S, i.e. an image, onto the ground plane, where $s \in S$ denotes a single pixel. Then, we project each LIDAR point to this lattice, which has a label terrain (blue in Fig. 1(a)), building roof (purple) or vehicle (black). This projection results in a 2D class label map (Fig. 1(a)) and an intensity map (Fig. 1(b)), where multiple point projections to the same pixel are handled by a point selection algorithm, which gives higher precedence to vehicle point candidates. Since the projection of the sparse point cloud to a regular image lattice may result in pixels without definite class labels and intensities, we also use *undefined labels* at certain pixels (white pixels of Fig. 1(a) and (e)). Terrain and roof regions are jointly referred as background in the following.

3. L^2-Marked Point Process Model

The inputs of this step are the label and intensity maps over the pixel lattice S, which were extracted in the previous section. The detection is mainly based on the label map, but additional evidences are extracted from the intensity image, where several cars appear as salient bright blobs due to their shiny surfaces. We assume that each vehicle u can be approximated from top view by a rectangle, which is described by five parameters: c_x and c_y center coordinates in the lattice S, e_L, e_l side lengths and θ orientation (Fig. 1(c)). Let be $R_u \subset S$ the set of pixels corresponding to u. Note that with replacing the rectangle shapes for parallelograms, the "shearing effect" of moving vehicles may also be modeled [6], but in the considered test data this phenomenon could not be reliably observed. Let $\mathcal{H}$ be the space of u objects. We define a neighborhood relation $\sim$ in $\mathcal{H}$: $u \sim v$ iff the distance of the object centers is smaller than a threshold. We describe the scene by a Two-level Marked Point Process (L^{2}MPP) model: a global configuration ω is the set of k traffic segments, $\omega = \{\psi_1, \ldots, \psi_k\}$, where each traffic segment ψ_i ($i = 1 \ldots k$) is a configuration of n_i vehicles, $\psi_i = \{u_1^i, \ldots, u_{n_i}^i\} \in \mathcal{H}^{n_i}$. Here we prescribe that $\psi_i \cap \psi_j = \emptyset$ for $i \neq j$, while the k set number and $n_1, \ldots, n_k$ set cardinality values may be arbitrary (and initially unknown) integers. We mark with $u \prec \omega$ if u belongs to any ψ in ω, i.e. $\exists \psi_i \in \omega : u \in \psi_i$. Ω denotes the space of all the possible ω global configurations. Taking an inverse approach, an energy function $\Phi(\omega)$ is defined, which can evaluate each $\omega \in \Omega$ configuration based on the observed data and prior knowledge. Therefore, the energy can be decomposed into a data term and a prior term: $\Phi(\omega) = \Phi_d(\omega) + \Phi_p(\omega)$, and the optimal ω is obtained by minimizing $\Phi(\omega)$.

3.1 Data-dependent energy terms

Data terms evaluate the proposed u vehicle candidates based on the input label- or intensity maps, but independently of other objects of the population. The data modeling process consists of two steps. *First*, we define different $f(u) : \mathcal{H} \to \mathbb{R}$ features which evaluate a vehicle hypothesis for u in the image, so that 'high' $f(u)$ values correspond to efficient vehicle candidates. The following features are utilized by our model:

- $f^{\mathrm{ve}}(u)$ *vehicle evidence* feature: the ratio of the number of vehicle classified pixels within the proposed rectangle R_u of object u (see Fig. 1(d)-(e))
- $f^{\mathrm{eb}}(u)$ *external background* feature: the ratio of background classified pixels around the proposed rectangle, within the $T^{(i)}$ *external* regions of Fig. 1(d)
- $f^{\mathrm{it}}(u)$ *intensity* feature: ratio of the vehicle colored pixels within R_u (Fig. 1(f))

In the *second step*, we construct $\varphi_d^f(u)$ *data driven* energy subterms for each feature f, by attempting to satisfy $\varphi_d^f(u) < 0$ for real objects and $\varphi_d^f(u) > 0$ for false candidates. For this purpose, we project the feature domain to $[-1, 1]$ with a monotonously decreasing nonlinear $\mathcal{Q}(f, d_0^f)$ function [3, 2], whose zero value is equal to parameter d_0^f. With other words, d_0^f is the object acceptance threshold for feature f, which can be set based on manually annotated training data in a straightforward way. Once we obtained the $\varphi_d^{\mathrm{ve}}(u)$, $\varphi_d^{\mathrm{eb}}(u)$, $\varphi_d^{\mathrm{it}}(u)$ subterms, the joint data energy of object u is derived as $\varphi_d(u) = \max(\min(\varphi_d^{\mathrm{it}}(u), \varphi_d^{\mathrm{ve}}(u)), \varphi_d^{\mathrm{eb}}(u))$.

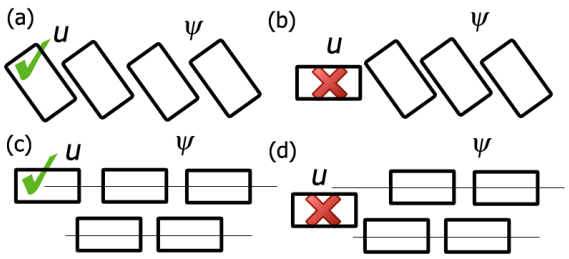

Figure 2. Favored (√) and penalized (×) sub-configurations within a traffic segm.

Here the min and max operators are equivalent to the logical OR resp. AND operations for the different feature constraints in the negative fitness domain. We do not prescribe simultaneously the *vehicle evidence* and *intensity* constraints, since usually not all vehicles appear as bright blobs in the intensity map. The data term of the ω configuration is obtained as the sum of the individual object energies: $\Phi_d(\omega) = \sum_{u \prec \omega} \varphi_d(u)$.

3.2 Prior terms

The prior terms implement geometric constraints between different objects and traffic segments of ω.

$$\Phi_p(\omega) = \sum_{\substack{u,v \prec \omega \\ u \sim v}} I(u,v) + \sum_{u \prec \omega, \psi \in \omega} A(u,\psi) \quad (1)$$

where $I(u,v)$ penalizes any overlapping rectangles within the ω configuration $I(u,v) = \frac{\text{Area}\{R_u \cap R_v\}}{\text{Area}\{R_u \cup R_v\}}$.

To measure if a vehicle u is appropriately arranged with respect to a traffic segment ψ, we define an alignment distance measure $d_\psi(u) \in [0,1]$ which is the average of two terms: *firstly*, the normalized angle difference between u and the mean angle within ψ (see Fig. 2(a)-(b)), *secondly*, with using RANSAC, we fit one or a couple of parallel lines to the object centers within ψ, and calculate the normalized distance of the center of u from the closest line (Fig. 2(c)-(d)). For prescribing spatially connected traffic segments, we use a constant high difference factor, if u has no neighbors within ψ w.r.t. relation $\sim$. Thus we derive a modified distance:

$$\hat{d}_\psi(u) = \begin{cases} 1 & \text{if } \nexists v \in \psi \backslash \{u\} : u \sim v \\ d_\psi(u) & \text{otherwise} \end{cases}$$

We define the $A(u,\psi)$ arrangement term of (1) in the following way. We slightly penalize vehicle groups which only contain a single vehicle: with a small $0 < c \ll 1$ constant $A(u,\psi) = c$ iff $\psi = \{u\}$. Otherwise, large $\hat{d}_\psi(u)$ is *penalized* if $u \in \psi$; and *favored* if $u \notin \psi$:

$$A(u,\psi) = \mathbf{1}_{u \in \psi} \cdot \hat{d}_\psi(u) + \mathbf{1}_{u \notin \psi} \cdot (1 - \hat{d}_\psi(u))$$

where $\mathbf{1}_E \in \{0,1\}$ is an indicator function of event E.

4 Optimization

To estimate the optimal object configuration, we have proposed a two-level modification of the Multiple Birth and Death Algorithm [1], as follows:

Initialization: start with empty population $\omega = \emptyset$, set the birth rate b_0, initialize the inverse temperature parameter $\beta = \beta_0$ and the discretization step $\delta = \delta_0$.

Main program: alternate the following three steps:

• *Birth step*: Visit all pixels on the image lattice S one after another. At each pixel s, with probability δb_0, generate a new object u with center s and random e_L, e_l and θ parameters. For each new object u, with a probability $p_u^0 = \mathbf{1}_{\omega=\emptyset} + \mathbf{1}_{\omega \neq \emptyset} \cdot \min_{\psi_j \in \omega} \hat{d}_{\psi_j}(u)$, generate a new ψ empty traffic segment, add u to ψ and ψ to ω. Otherwise, add u to an existing traffic segment $\psi_i \in \omega$ with a prob. $p_u^i = (1 - \hat{d}_{\psi_i}(u)) / \sum_{\psi_j \in \omega} (1 - \hat{d}_{\psi_j}(u))$.

• *Death step*: Consider the actual configuration of all objects within ω and sort it by decreasing values depending on $\varphi_d(u) + A(u,\psi)\big|_{u \in \psi}$. For each object u taken in this order, compute $\Delta\Phi_\omega(u) = \Phi_\mathcal{D}(\omega / \{u\}) - \Phi_\mathcal{D}(\omega)$, derive the *death rate* $d_\omega(u)$ as

$$d_\omega(u) = \Gamma(\Delta\Phi_\omega(u)) = \frac{\delta \exp(-\beta \cdot \Delta\Phi_\omega(u))}{1 + \delta \exp(-\beta \cdot \Delta\Phi_\omega(u))},$$

and delete object u with probability $d_\omega(u)$. Remove empty traffic segments from ω, if they appear.

• *Group re-arrangement*: Propose randomly group merge, group split and vehicle re-clustering moves. For each proposed move $\mathbf{M}$, calculate the corresponding energy cost $\Delta\Phi_\omega^\mathbf{M}$, and apply the move with a probability $\Gamma(\Delta\Phi_\omega^\mathbf{M})$, similarly to the case in the death step.

Convergence test: if the process has not converged yet, increase β and decrease δ with a geometric scheme, and go back to the birth step.

5 Evaluation

We evaluated our method in four aerial LIDAR data sets (provided by Astrium GEO-Inf. Services - Hungary), which are captured above crowded urban areas and contain in aggregate 471 vehicles. The parameters of the method were set based on a limited number of training samples, similarly to [1]. For accurate Ground Truth (GT) generation, we have developed an accessory program with graphical user interface, which enables us to manually create and edit a GT configuration of rectangles. We have performed quantitative evaluation both at object and at pixel levels. At object level, we have counted the number of true positive, true negative and false positive objects. Then with considering the Number of real Vehicles (NV), we calculated the F-rate of

Table 1. Obj. and pix. level F-rates (in %) by the DP [4], hX [5] and the proposed L^{2}MPP (^{2}M) methods, and the Group Classification Rate (GR) of the L^{2}MPP model.

Set	NV*	Object level %			Pixel level %			GR
		DP	hX	^{2}M	DP	hX	^{2}M	^{2}M
#1	78	78	68	**96**	64	46	**89**	**94**
#2	91	90	93	**98**	77	77	**88**	**93**
#3	132	70	74	**83**	61	46	**66**	**86**
#4	170	85	87	**89**	77	76	**64**	**92**
All	471	83	82	**91**	70	61	**80**	**91**

*NV = Number of real Vehicles in the test set

Figure 3. Detection result with four clusters. Vehicles of different segments are displayed with different colors, background is interpolated for visualization.

the detection (harmonic mean of precision and recall). At pixel level, we compared the vehicle silhouette mask to the GT mask, and calculated the F-rate of the match [1]. We have also measured the correct Group Classification Rate (GR, %) among the true positive samples, considering GT classification of human observers.

For comparison, we have selected a grid-cell-based algorithm from [4], called *DEM-PCA* (DP); and a recent state of the art method [5], which uses h-maXima (hX) transform followed by watershed segmentation. Some qualitative results are shown in Fig. 3 and 4 (best viewed in color), and the quantitative evaluation is provided in Table 1. Since the reference methods do not deal with vehicle grouping, only the car detection rates are compared: the proposed L^{2}MPP model surpasses the references both at object and at pixel levels.

6. Conclusions and future work

This paper has proposed a novel Two-Level MPP model for joint extraction of vehicles and traffic segments in aerial point cloud data. The efficiency of the approach has been tested with real-world LIDAR measurements, and its advantages versus two reference methods have been demonstrated. Note that in the proposed model, the vehicles are grouped based on similar orientation, but we have experienced that the method can deal with car groups on slightly curved roads as well. As future work, we plan to extend the prior terms of our method to handle more complex vehicle arrangement patterns such as strongly curved exit ramps or roundabouts.

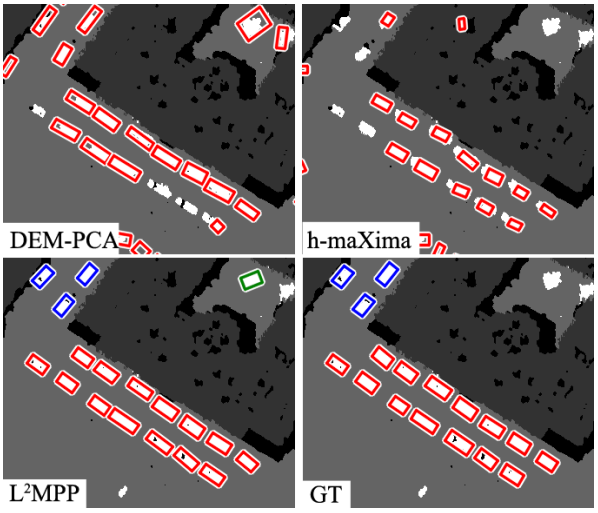

Figure 4. Method comparison on a sample

References

[1] C. Benedek, X. Descombes, and J. Zerubia. Building development monitoring in multitemporal remotely sensed image pairs with stochastic birth-death dynamics. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(1):33–50, 2012.

[2] A. Börcs and C. Benedek. A marked point process model for vehicle detection in aerial lidar point clouds. In *ISPRS Congress*, Melbourne, Australia, 2012.

[3] F. Lafarge, G. Gimel'farb, and X. Descombes. Geometric feature extraction by a multi-marked point process. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(9):1597–1609, 2010.

[4] Á. Rakusz, T. Lovas, and Á. Barsi. Lidar-based vehicle segmentation. *International Archives of Photogrammetry and Remote Sensing*, XXXV(2):156–159, 2004.

[5] W. Yao, S. Hinz, and U. Stilla. Automatic vehicle extraction from airborne lidar data of urban areas aided by geodesic morphology. *Pattern Recogn. Letters*, 31(10):1100 – 1108, 2010.

[6] W. Yao and U. Stilla. Comparison of two methods for vehicle extraction from airborne lidar data toward motion analysis. *IEEE Geoscience and Remote Sensing Letters*, 8(4):607–611, 2011.