A Minimum Cover Approach for Extracting the Road Network from Airborne LIDAR Data

Qihui Zhu
University of Pennsylvania
Philadelphia, PA, USA
qihuizhu@seas.upenn.edu

Philippos Mordohai
Stevens Institute of Technology
Hoboken, NJ, USA
mordohai@cs.stevens.edu

Abstract

We address the problem of extracting the road network from large-scale range datasets. Our approach is fully automatic and does not require any inputs other than depth and intensity measurements from the range sensor. Road extraction is important because it provides contextual information for scene analysis and enables automatic content generation for geographic information systems (GIS). In addition to these two applications, road extraction is an intriguing detection problem because robust detection requires integration of local and long-range constraints. Our approach segments the data based on both edge and region properties and then extracts roads using hypothesis testing. Road extraction is formulated as a minimum cover problem, whose approximate solutions can be computed efficiently. Besides detecting and extracting the road network, we also present a technique for segmenting the entire city into blocks. We show experimental results on large-scale data that cover a large part of a city, with diverse landscapes and road types.

1. Introduction

Automatic road extraction has recently attracted the attention of the creators and providers of GIS content in traditional formats, as well as over the Internet and wireless networks. This is due to the maturation of sensor and analysis technology and increased consumer interest. Apart from commercial applications, the extracted road network can be used for urban and emergency response planning. Progress has been made since the early work of [1, 17], but the problem is far from being solved. In 2007, the DARPA Urban Challenge [3, 15, 23] demonstrated autonomous navigation, but road locations where provided in GPS coordinates to the participants. We present an approach for road detection from airborne Light Detection And Ranging (LIDAR) data that is applicable to large-scale datasets that is offline, but fully automatic.

Besides the practical considerations mentioned above, there are several reasons that make this problem attractive from a scientific point of view. The consensus regarding modern scene understanding is that much can be gained by the use of contextual relationships between the environment and the objects. The road network provides rich contextual information for both urban and rural environments, but its extraction is not without difficulties. Some of these difficulties are due to local extraction failures, because of occlusion or illumination variations, while others are due to the non-local nature of the constraints that need to be enforced. Besides being long-range, many of the constraints that are desirable for roads, are often violated. For example, roads are in general straight and elongated. They intersect at right angles and they are parallel to other roads. When these constraints hold, they can aid the extraction process significantly. For instance, Price [20] extracts the road grid from urban imagery ensuring that its topology is correct. The system was very accurate for cities with "stereotypical midwestern US patterns", but roads that deviated from these patterns were not detected. The strict a priori enforcement of other constraints can lead to similar behaviors. Our approach integrates the constraints into early processing stages, but allows significant deviations from them when other evidence suggests the presence of roads with unexpected orientations or large curved segments.

Our method operates on range data captured from airborne LIDAR sensors utilizing both the 3D measurements and the intensity response of the sensors. The latter is a measure of the strength of the return pulse and is a function of distance, but also of the material reflecting the laser. The benefit of using the intensity of the LIDAR over regular electro-optic (EO) imagery is that we do not need additional sensors and, most importantly, that it is not affected by illumination variations and shadows. This property makes the appearance of materials fairly consistent despite the presence of shadows. The disadvantage of LIDAR data compared to images is lower resolution.

Our segmentation approach combines region and edge information and is able to enforce desirable properties during the extraction and not as post-processing steps (Section 4). A fundamental characteristic that distinguishes roads from other asphalt-paved surfaces, such as parking lots, is elongation. Unlike previous methods, our approach does not classify regions as potential roads and then searches for elongated structures among these regions. The minimum cover formulation [8] of the problem imposes a preference for a set of long, straight segments that explains the heat map of road likelihood while incurring minimum cost.

In addition, we show how the road network can be used in a hierarchical framework for analyzing urban datasets. Specifically, we use the computed road likelihood to segment city blocks. This provides a natural way to divide very large urban datasets into manageable yet semantical meaningful blocks. Subsequent analysis can then focus on each block separately in order to detect buildings and smaller objects. Our approach is based on the Normalized Cuts framework [21] that considers *intervening contours* between points [13] and is able to segment blocks even if road detection has missed segments of the road.

2. Related Work

In this section, we briefly review related work on road extraction beginning with approaches that require a road database and aim at updating it. Representative work on road database updating includes that of Baltsavias and Zhang [2]. The proposed system operates on images, digital terrain models (DTMs) and geospatial databases. It fuses multiple cues and relies on detailed models and rules for different types of roads. Grote et al. [9] also aim at assessing road databases. The normalized cuts algorithm [21] is used to segment the image using color, edges and road color information. The latter can be learned from the images since road centerlines are provided by the database.

A more challenging problem than road database assessment is road extraction from images without the aid of a database. Laptev et al. [12] addressed road extraction in scale-space by modeling roads as curves at coarse scales and ribbons at finer scales, achieving good performance on rural scenes. Hinz and Baumgartner [10] advocate a knowledge-based approach with detailed modeling of road categories that operates on multiple images and DTMs. Youn et al. [26] exploit the distribution of edges in the image to search for roads along the dominant edge directions. A sequence of hierarchical clustering steps followed by adaptive snakes is employed to detect likely roads and intersections. Stoica et al. [22] formulate the problem of extracting the road network using Gibbs point processes. Roads are treated as lines with no width, but complex relationships and constraints between them can be modeled using a Markov Chain Monte Carlo scheme. Porikli et al. [18] consider low, mid and high-level features in a learning approach implemented in the form of a multi-layer neural network. Dal Poz et al. [6] propose a rule-based system that initially forms straight road fragments by chaining road segments. Proximity and collinearity are then used to complete partial detections. Mayer et al. [14] compared six methods for automatic road extraction on a benchmark of high-resolution aerial and satellite images. The methods consider either edges or regions, but not both. The conclusion from this study was that good performance can be attained automatically for medium complexity rural scenes, but not for dense urban areas. Moreover, many of the techniques do not scale up to datasets of realistic sizes.

We now turn our attention to methods that use LIDAR as the main input modality. Hu et al. [11] rely on LIDAR data using images for verification. They segment the LIDAR data according to intensity and use a hypothesize and test scheme that employs iterative Hough transforms and topology analysis to favor a grid structure. Results on a dataset of 10.6 million LIDAR points are shown. Clode et al. [5] presented a region growing approach for road extraction from airborne LIDAR data. It can be combined with building and tree detectors to improve the accuracy of the extraction. The region-growing process, however, makes decision at the pixel level and cannot favor elongated regions. Poullis et al. [19] present an approach that bears some similarities with our work. They use Gabor filters to detect local edges and tensor voting to infer longer contours based on the support they receive from their neighbors. Graph cuts [4] are used to segment the image according to the dominant contour orientation at each patch. Finally, the outputs of filters that respond to road boundaries and interiors are used as input to an iterative Hough transform that extracts straight road segments.

3. Overview of the Approach

This section provides a summary of the processing steps, which can also be seen in Algorithm 1. We start by extracting ground planes from 3D point cloud data using a simple segmentation technique (Section 4.1). An aspect of this step worth emphasizing is that our algorithm does not extract a single ground plane, but is able to extract multiple approximately horizontal planes on which roads may exist. This enables the detection of roads at different elevations. Such situations are encountered in areas with steep cliffs, where a road may run at the bottom of the cliff while another road may run at the top. See Fig. 4(b) for an example.

The ground planes cover all the roads, but typically contain many other objects such as grass, sidewalks and parking lots. Height, however, is not a good cue for further discrimination. Therefore, once the ground has been identified, 3D points of the ground planes are projected to generate a 2D map of LIDAR intensity values. The task of road extraction

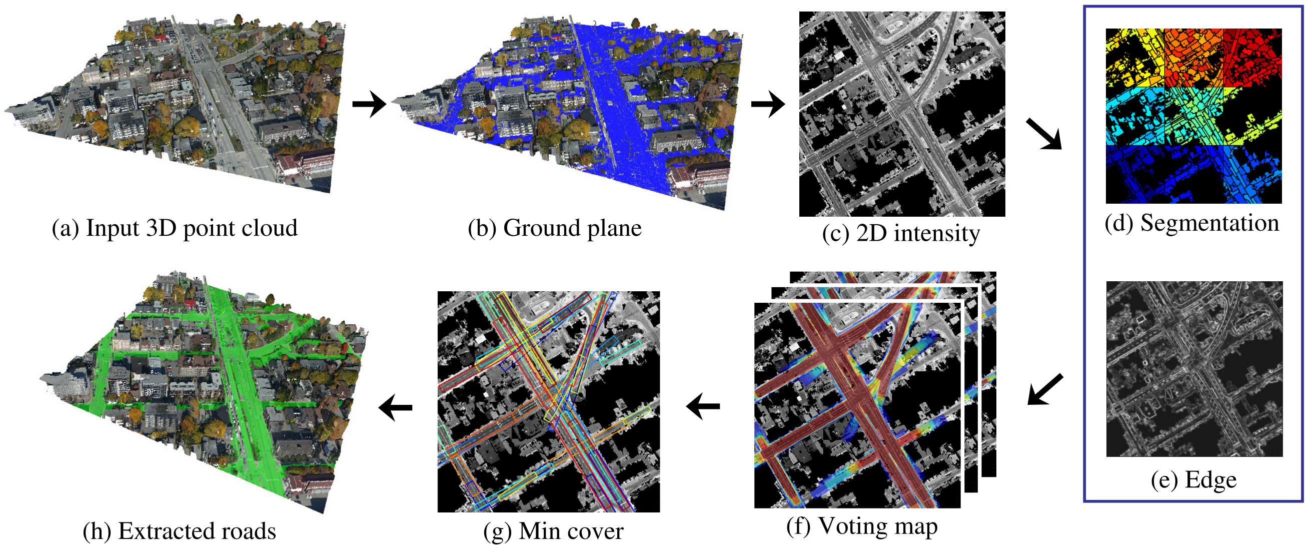


Figure 1. System overview. Given input 3D point cloud (a), our system first extracts multiple ground planes marked in blue in (b). The ground planes contain roads mixed with other objects such as grass, sidewalks, parking lots, playgrounds, and possibly a few errors on low buildings. All the ground points are then projected onto a 2D image (c) with their laser intensities. Region segmentation (d) and edge detection (e) are performed on the image to generate boundary and interior features for roads. These local features are integrated into multiple heat maps (f) representing the likelihood of roads with different widths. A minimum cover algorithm extracts the salient road regions to explain the heat maps and automatically select the road widths, as shown in (g). Road points are transformed back to 3D and the final road extraction result is shown in (h). This figure is best viewed in color.

is now reduced to a search over these 2D maps.

In order to process very large datasets, we divide them in square tiles which are small enough so that computational cost is low, but large enough so that the property of roads to be elongated can be captured. We have found that $300m \times 300m$ tiles provide a good trade-off between these requirements. We divide the data in such tiles with significant overlaps and focus on extracting the roads correctly at the center of each tile. In practice, the tiles overlap their neighbors by $100m$ on each side.

We then perform segmentation on the tiles to obtain a set of segments on which we reason (Section 4.2). Two types of road features, *boundary features* and *interior features* are generated from both segmentation boundaries and edge detection on the 2D map. We accumulate these features to vote for rectangle hypotheses for roads, and combine them to obtain a heat map of road likelihood (Section 4.3).

Finally, a *minimum cover* algorithm is used to find a set of road segments which best cover this likelihood map (Section 4.4). The road widths are selected automatically via competitions of segments with different widths in the minimum cover process.

4. Road Extraction

In this section, we present our approach for road extraction in each tile.

Algorithm 1 Road Extraction

```
Extract multiple ground planes.
Project intensities of ground points to 2D maps $\mathcal{I}$.
Detect edges and perform region segmentation on $\mathcal{I}$ using [16, 21].
Compute the strengths of boundary and interior features using Eq. (1) and Eq. (3).
Generate road likelihood map $\mathcal{L}$ using Eq. (5).
Extract dominant road regions with the correct widths from $\mathcal{L}$ using minimum cover (Algorithm 2).
Transform the road regions back to 3D points for visualization.
```

4.1. Ground Plane Detection

The first step is to extract all possible ground planes. This is necessary since our datasets, and many areas in the world, contain multi-level terrain. We use a simple clustering procedure based on [7] that groups LIDAR points according to their x, y, z coordinates. The result is a large number of clusters, typically several thousand, for each tile. This grouping allows us to compute properties for each cluster and classify them as potential parts of a ground plane.

Given clusters of 3D points $\mathcal{C} = \{C_1, C_2, ..., C_n\}$, we can classify them as potential ground clusters $\mathcal{C}_{ground}$ according to the following criteria:

- *Planarity.* Compute PCA of the points in C_i. Let $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$ be the eigenvalues. If $\lambda_3/(\lambda_1+\lambda_2+\lambda_3) \geq T_{planar}$, prune C_i. This removes all the non-planar segments. In practice, we set $T_{planar} = 0.01$.
- *Vertical normal.* Let $\mathbf{u_3}$ be the smallest eigenvector of PCA of the points. If $\|(0,0,1)^T \cdot \mathbf{u_3}\| \leq T_{vert}$, prune C_i. This removes all planar segments that are not horizontal. We use $T_{vert} = 0.98$, which allows roads with a slope up to $11.5°$.

After pruning clusters that cannot be parts of the ground, we extract ground surfaces by grouping adjacent clusters in $\mathcal{C}_{ground}$. These surfaces are further pruned by removing those that are too small. Since height variations are small in these surfaces, the z coordinate is dropped and 2D LIDAR intensity images are generated with a resolution of $0.5m$ per pixel. The next processing steps are applied on these images.

4.2. Road Feature Extraction

Shape and appearance are both essential measurements for classifying road regions. Here, we describe the features that are used in Section 4.3 to generate road hypotheses. To capture the shape of long and straight road boundaries, we introduce the *boundary feature* computed by convolution with a polarized rectangular filter. We have also noticed that the texture of road regions typically has a strong regularity which can be identified by the *interior feature*. The two types of features are defined as follows.

Boundary feature. Road segments are bounded by long and straight lines. Hence we define a 2D boundary filter as:

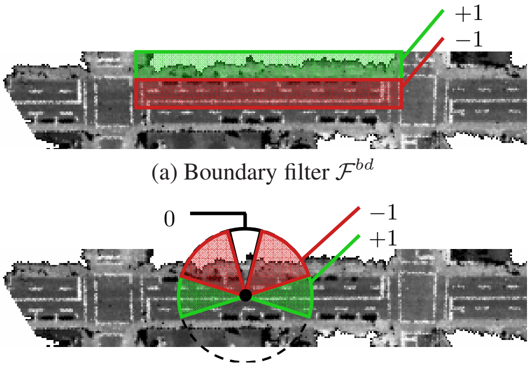
$$\mathcal{F}^{bd}(x,y) = \begin{cases} 1, & \text{if } (-w_1 \leq y \leq 0) \wedge (0 \leq x \leq \ell_1) \\ -1, & \text{if } (0 \leq y \leq w_1) \wedge (0 \leq x \leq \ell_1) \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

where (x,y) are the pixel coordinates, ℓ_1 and w_1 are the length and width of the filter. The boundary filter F_{bd} is similar to the canonical edge filter [16], but tuned to detect long and straight boundaries (see Fig. 2). The polarity of this filter assumes that the laser intensity on the road is lower than its surroundings (road is darker). Road boundaries can be detected by convolving the intensity image $\mathcal{I}$ with $\mathcal{F}^{bd}$:

$$S^{bd}(x,y) = \mathcal{F}^{bd} * \mathcal{I}(x,y) \tag{2}$$

We choose filtering as opposed to line fitting to give more tolerance to curved roads. We construct the mirror image of $\mathcal{F}^{bd}$ to detect the other side of the road boundary.

Interior feature. The agreement of interior edge orientations to the road direction is also an important feature. The interior edges are mainly caused by markings and separators on the road surface, which are approximately parallel to the road orientation. Therefore, we introduce the following filter to measure the strength of these edges:



(a) Boundary filter $\mathcal{F}^{bd}$

(b) Interior filter $\mathcal{F}^{int}$

Figure 2. Boundary and interior features. (a) superimposes the filter onto a road segment, where the green region of the filter has a value of $+1$ and value of red region is -1. Similarly, (b) shows the value of $\mathcal{F}^{int}$ w.r.t. edge orientation θ. Note that orientations almost orthogonal to the road direction have value 0, minimizing the effect of the crosswalk markings shown in the image.

$$\mathcal{F}^{int}_{\theta}(x,y) = \begin{cases} 1, & \text{if } (\min(\theta,\pi-\theta) \leq \theta_2) \wedge (x,y) \in B \\ -1, & \text{if } (\theta_2 < \min(\theta,\pi-\theta) < \pi/2-\theta_2) \wedge \\ & \quad (x,y) \in B \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

Here $\theta \in [0,\pi]$ is the edge orientation, $\theta_2 = 15°$ is the tolerance and $B = \{(x,y) : (|y| \leq w_2) \wedge (0 \leq x \leq \ell_2)\}$ is the rectangular support of the filter (see Fig. 2). Similar to boundary feature, the induced strength is measured by convolution

$$S^{int}(x,y) = \sum_{\theta} \mathcal{F}^{int}_{\theta} * \mathcal{E}(x,y,\theta) \tag{4}$$

where $\mathcal{E}(x,y,\theta)$ is the edge map. Edges are detected using [16] which outputs both the orientation and strength of edge pixels. Note that we do not penalize edges whose orientations are orthogonal to the road direction (hence $\mathcal{F}^{int}_{\theta}(x,y) = 0$) to tolerate intersections and crosswalks.

4.3. Road Hypotheses Generation

We generate road hypotheses from the boundary and interior features described above. To balance accuracy and efficiency, we sample hypotheses from region segment boundaries instead of edges or all image pixels. Region segmentation, such as the normalized cut algorithm (NCut) [21], can often capture faint boundaries missing from simple edge detection. We take the laser intensity image as input and apply NCut on it using intensity similarity as the only feature. The image is over-segmented to 100-200 regions in order to recall most of the road boundaries. As seen in Fig. 1, these region boundaries provide a set of much sparser yet accurate hypotheses compared to edges.

Road hypotheses are parameterized as rectangular windows $\mathcal{H} = \{H_i(x_i, y_i, \theta_i, w_i, \ell_i), i = 1, ..., n\}$, where (x_i, y_i), θ_i, w_i and ℓ_i are top-left corner, orientation, width and length of window H_i. Corners of windows (x_i, y_i) are sampled from long and straight segmentation boundaries. The orientations θ_i are obtained from the local maxima in the histogram of edge orientations weighted by edge magnitude. The width w_i is sampled from $10m$ to $25m$ and the length l_i is sampled from $40m$ to $300m$ in our experiments.

For each window H_i, we compute its boundary and interior feature strength by convolving with filters $\mathcal{F}^{bd}(H_i)$ and $\mathcal{F}_{\theta}^{int}(H_i)$ using Eq. (2) and (4), with the window H_i as support. Images are rotated by different angles θ_i before applying these filters. Convolutions on a large set of rectangular windows can then be computed efficiently using integral images [25]. Only windows with S^{bd} and S^{int} both higher than certain thresholds are considered as valid hypotheses. Windows with similar width k form a hypothesis set $\mathcal{H}_k^{valid} = \{H_i \mid S_{H_i}^{bd}(x, y) \geq T_{bd}, S_{H_i}^{int}(x, y) \geq T_{int}\}$. Each valid window casts a vote for the pixels it covers. Votes are accumulated in a likelihood map:

$$\mathcal{L}_k(x, y) = \sum_{H_i \in \mathcal{H}_k^{valid}} \beta_{bd} S_i^{bd}(x, y) + \beta_{int} S_i^{int}(x, y) \quad (5)$$

where $S_i^{bd}(x, y)$ and $S_i^{int}(x, y)$ are the boundary and interior feature strength induced by hypothesis H_i. The coefficients β_{bd} and β_{int} balance the two scores. The likelihood $\mathcal{L}(x, y)$ is further capped by a threshold $\mathcal{L}_{max}$, above which we are confident that (x, y) belongs to the road and higher likelihood does not make any difference.

To select road width in the later process, we generate separate likelihood maps $\mathcal{L}_k$ from hypothesis windows with different widths. Each heat map contains road hypotheses of all orientations (see Fig. 1(f)). The advantage of combining all the orientations into a single heat map is that intersections will gain more support from roads connecting to them. This integrates information from long range and alleviates the problem that local features tend to miss intersections which are not elongated.

4.4. Final Detection as Minimum Cover

The final detection step is to detect road regions by combining hypotheses generated from the previous step. Key issues of this step include 1) determining the extent of road regions, *i.e.* distinguishing the cases of road termination vs. broken road hypotheses due to corrupted signals or occlusion; 2) consolidating multiple road hypotheses with different orientations; and 3) selecting the correct road width. Addressing these problems requires long-range information integration in addition to local road features.

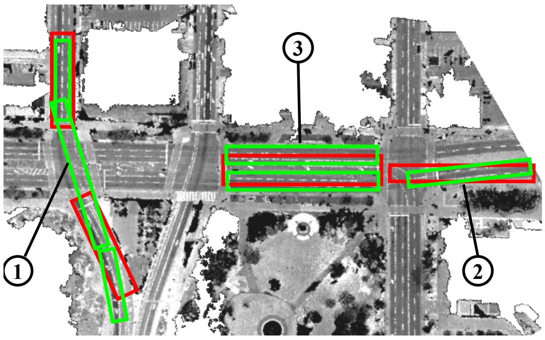


Figure 3. Minimum cover. There are three typical scenarios, in which green rectangles are chosen over red rectangles: 1) determine whether to terminate or not; 2) different orientation; 3) selecting road width. In 3), there is an ambiguity of splitting the road into two lanes, or merging them.

An intuitive idea is to explain the likelihood maps by the sparsest set of road segments. We formulate this idea as covering the likelihood maps $\mathcal{L}_k$ using a set of road hypothesis windows with the minimum cost. This is known as the *minimum cover problem* [24] in which there is a universe of elements, a collection of subsets and a cost function defined on the subsets. One would like to find the subcollection from these subsets covering the elements with the minimum total cost. In our setting, the sets and elements are:

- *Elements.* We generate seeds $\mathcal{S} = \{s_j\}$ by sampling the rectangle centerlines of road hypotheses. The sampled points are chosen instead of pixels in the likelihood map for computational efficiency.
- *Sets.* Rectangular hypotheses $\mathcal{R} = \{R_1, ..., R_n\}$ from the hypothesis generation step. We also create dummy sets which cover only a single seed.

Each one of the hypotheses R_i has a cost computed by:

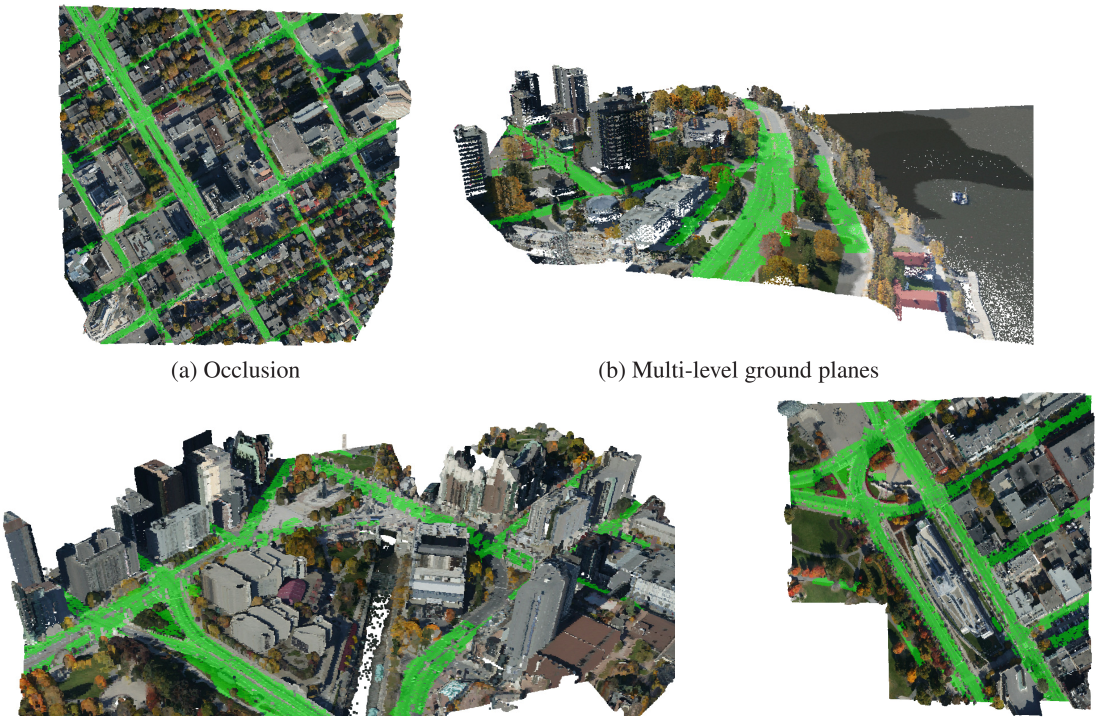
$$Cost(R_i) = \sum_{s_j \in R_i} C(s_j) + A \quad (6)$$

where $C(s_j) = 1 - \mathcal{L}(s_j)/\mathcal{L}_{max}$ measures the cost of covering a seed s_j. The constant A favors choosing fewer rectangles. We seek to extract a subset $\mathcal{R}_\mathcal{S} \subseteq \mathcal{R}$ minimizing:

$$Cost_{cover}(\mathcal{R}_\mathcal{S}) = \sum_{R_i \in \mathcal{R}_\mathcal{S}} Cost(R_i) + \sum_{s_j \notin \mathcal{R}_\mathcal{S}} \frac{\mathcal{L}(s_j)}{\mathcal{L}_{max}} \quad (7)$$

The first summation computes the cost of covering by $\mathcal{R}_\mathcal{S}$. The second summation counts the loss of missing seeds, or equivalently the cost from the dummy sets. The result will cover as many good seeds as possible by competition of overlapping hypotheses (see Fig. 3).

The minimum cover problem is NP-hard in general, but a greedy algorithm can achieve a solution within a constant



(a) Occlusion

(b) Multi-level ground planes

(c) Non-orthogonal junctions

(d) Curved roads

Figure 4. Screenshots of road extraction results. Each image shows a view of a tile that presents particular challenges. The roads have been marked in green in all cases. From left to right, top to bottom the challenges are: (a) severe occlusion by trees and buildings; (b) roads on multiple level ground planes; (c)non-orthogonal junctions; (d) non-orthogonal roads and parking lots.

factor of the optimal cover [8]. We minimize the covering cost Eq. (7) by sequentially finding the hypothesis R_i with the lowest cost density:

$$Density(R_i) = \sum_{\substack{s_j \notin R_1,\ldots,R_{i-1} \\ s_j \in R_i}} \frac{C(s_j) + A}{|R_i \setminus \bigcup_{k=1}^{i-1} R_k|} \quad (8)$$

where $|R_i \setminus \bigcup_{k=1}^{i-1} R_k|$ is the number of new seeds covered by R_i. Recall that the constant A penalizes against solutions with extremely large numbers of very small rectangles. By adding a fixed cost to each rectangle, we can obtain solutions with small numbers of larger rectangles that cover areas of relatively higher cost, due to occlusion or noise. This allows us to enforce desirable properties, such as elongation. The process is terminated if all the seeds have been covered or the cost density $Density(R_i)$ is too high. The process is summarized in Algorithm 2.

Algorithm 2 Minimum cover

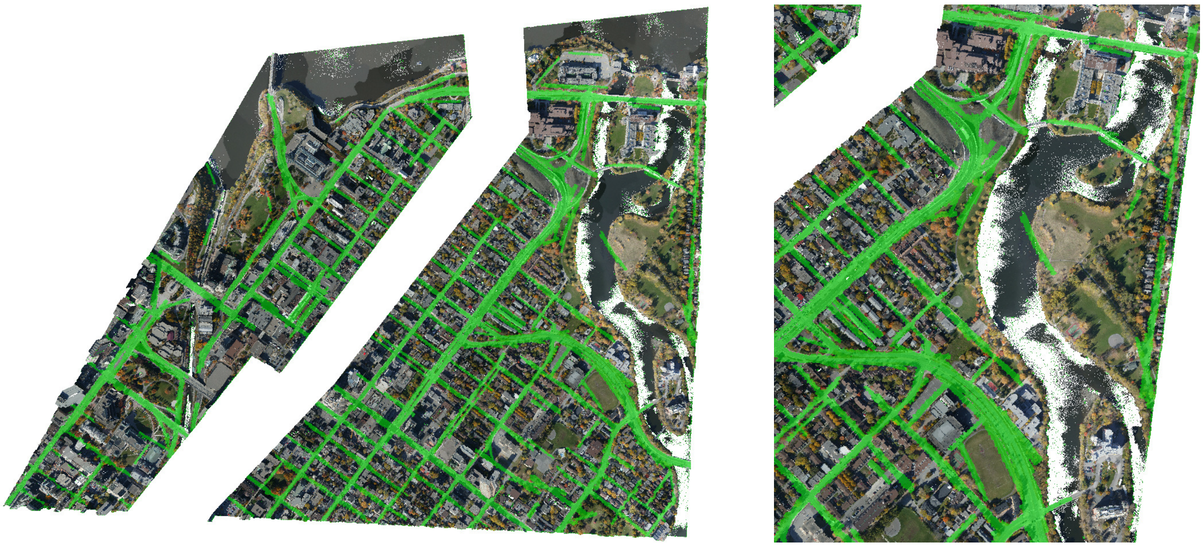
```
Generate seeds $\mathcal{S}$. Set $\mathcal{R}_{\mathcal{S}} = \emptyset$.
Repeat
    Find $R_i$ with the minimum density Eq. (8).
    Update the seeds covered by all the hypotheses.
Until seeds $\mathcal{S}$ are all covered or $Density(R_i) > D_{max}$.
```

4.5. Dataset and Results

We present results on a large point cloud (166 million points) that covers an area of $3km \times 3km$ in Ottawa, Canada. The data were collected by airborne LIDAR sensors and are publicly available[1]. Because the collection was performed in multiple flights, the dataset is inhomogeneous with varying density and sampling patterns. The dataset also includes points captured by terrestrial sensors. We do not use these points because data density would be sufficient to detect the roads on which the vehicle drove in a trivial way. No images are available, but each point has RGB values associated with it. These values are not used for processing, but only for visualization. A large part of the dataset is not available at the above website resulting in wide stripe of missing data that is visible in the figures.

Figure 4 shows screenshots of 3D models that present particular challenges to our method, as well as the extraction results. Figure 5 shows the extracted road network for the entire dataset. All results are generated with fixed settings for the parameters.

[1] $http : //www.daytaohio.com/Wright_{S}tate100.php$



(a) Extracted road network

(b) Detail

Figure 5. Screenshot of the road network for the entire point cloud that contains 166 million points and covers $3 \times 3km$. Our approach successfully extracts roads with varying width, large curved sections and bridges with limited false positives and missed detections.

5. City Block Segmentation

City blocks can be segmented using the extracted roads as boundaries, providing a natural partitioning of the data for further processing. The challenge is that the detected roads are not necessarily closed, therefore simple connected component or greedy region growing would create leakage on blocks. We use NCut [21] to segment the city blocks and borrow the idea of "intervening contours" from image segmentation [13] to incorporate roads. A graph consisting of projected 2D points is constructed and each point (x, y) has a road likelihood $\mathcal{L}(x, y)$ from the road extraction step. Graph nodes within a radius of $80m$ are connected. Edge weights between nodes i and j are defined by the highest road likelihood lying between them:

$$W(i, j) = \exp\{-\max_{(x,y)\in\ell(i,j)} \mathcal{L}(x, y)\} \tag{9}$$

where $\ell(i, j)$ denotes the straight line connecting nodes i and j. Hence, road likelihood, but not hard decisions on the presence of roads, is encoded in W. This allows us to obtain the most likely city blocks, even if some road segments are missing. We run NCut on W to generate a global segmentation. We have performed extensive tests on the Ottawa dataset and the results are very promising (see Fig. 6).

6. Conclusion

We have presented an approach for road detection based on spectral segmentation and an approximation to the minimum cover problem. The strengths of our approach are that it combines edge and region information in a single step and that it is able to select a relatively small number of hypotheses that explain the likelihood map. It can be easily extended to extract other ribbon-like structures by considering appropriate features and selecting an appropriate parameterization for the hypotheses.

The most urgent future work is a formal evaluation of the current results. We plan to undertake this manually-intensive task soon and provide results for completeness (recall) and correctness (precision) not only on road centerline detection, but also on width and regions of special interest such as junctions, partially occluded roads in dense, urban regions and faint, dirt roads in rural regions.

References

[1] R. Bajcsy and M. Tavakoli. Computer recognition of roads from satellite pictures. *IEEE Trans. Systems, Man and Cybernetics*, 6(9):623–637, 1976.

[2] E. Baltsavias and C. Zhang. Automated updating of road databases from aerial images. *International Journal of Applied Earth Observation and Geoinformation*, 2005.

[3] J. Bohren, T. Foote, J. Keller, A. Kushleyev, D. Lee, A. Stewart, P. Vernaza, J. Derenick, J. Spletzer, and B. Satterfield. Little ben: The ben franklin racing team's entry in the 2007 darpa urban challenge. *Journal of Field Robotics*, 25(9):598–614, 2008.

[4] Y. Boykov and M. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In *ICCV*, pages I: 105–112, 2001.

[5] S. Clode, F. Rottensteiner, and P. Kootsookos. Improving city model determination by using road detection from Lidar data. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2005.

[6] A. Dal Poz, R. Zanin, and G. do Vale. Automated extraction of road network from medium-and high-resolution images. *Pattern Recognition and Image Analysis*, 2006.

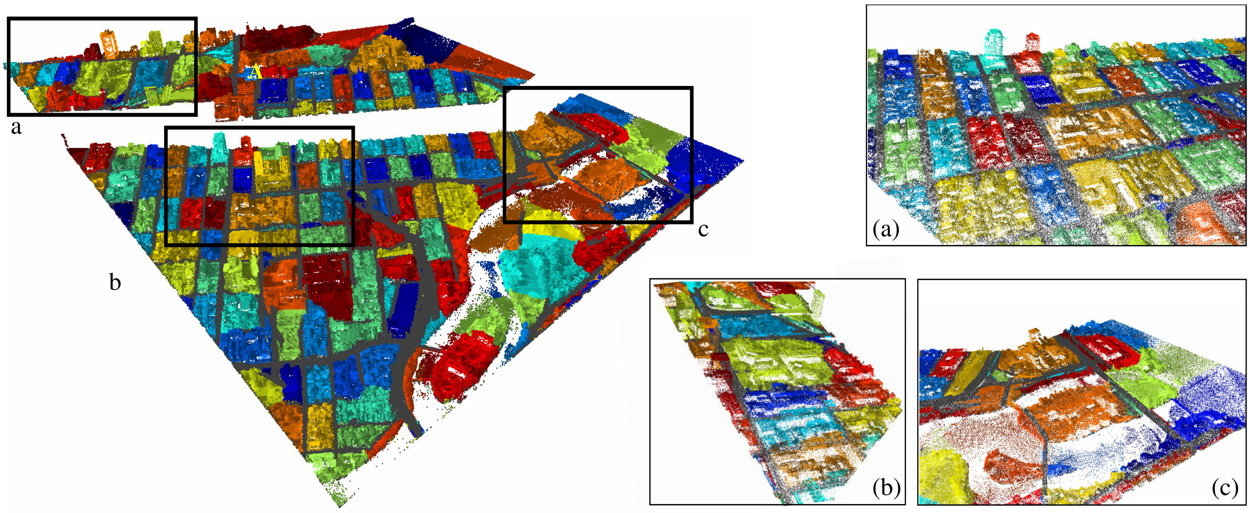


Figure 6. City block segmentation on Ottawa. The left figure shows the segmentation of city blocks on the entire city using the extracted roads. Extracted roads are marked in dark grey and city blocks are marked in distinct random colors. Zoom-in views of three regions are shown on the right. The segmentation is highly accurate in regular street blocks as well as irregular suburban regions. The 3D point clouds are subsampled by a rate of 8 for display purposes and the figure is best viewed in color.

[7] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 59(2):167–181, 2004.

[8] P. F. Felzenszwalb and D. McAllester. A min-cover approach for finding salient curves. In *Workshop on Perceptual Organization in Computer Vision*, 2006.

[9] A. Grote, M. Butenuth, and C. Heipke. Road extraction in suburban areas based on normalized cuts. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 36 (3/W49A)*, pages 51–56, 2007.

[10] S. Hinz and A. Baumgartner. Automatic extraction of urban road networks from multi-view aerial imagery. *ISPRS J. of Photogrammetry and Remote Sensing*, 58(1):83–98, 2003.

[11] X. Hu, C. Tao, and Y. Hu. Automatic road extraction from dense urban area by integrated processing of high resolution imagery and LIDAR data. In *ISPRS Congress*, 2004.

[12] I. Laptev, H. Mayer, T. Lindeberg, W. Eckstein, C. Steger, and A. Baumgartner. Automatic extraction of roads from aerial images based on scale space and snakes. *Machine Vision and Applications*, 12(1):23–31, 2000.

[13] T. Leung and J. Malik. Contour continuity in region-based image segmentation. In *ECCV*, page I: 544, 1998.

[14] H. Mayer, S. Hinz, U. Bacher, and E. Baltsavias. A test of automatic road extraction approaches. In *Photogrammetric Computer Vision*, 2006.

[15] M. Montemerlo, J. Becker, S. Bhat, H. Dahlkamp, D. Dolgov, S. Ettinger, D. Haehnel, T. Hilden, G. Hoffmann, B. Huhnke, D. Johnston, S. Klumpp, D. Langer, A. Levandowski, J. Levinson, J. Marcil, D. Orenstein, J. Paefgen, I. Penny, A. Petrovskaya, M. Pflueger, G. Stanek, D. Stavens, A. Vogt, and S. Thrun. Junior: The stanford entry in the urban challenge. *Journal of Field Robotics*, 25(9):569–597, 2008.

[16] M. C. Morrone and R. A. Owens. Feature detection from local energy. In *Pattern Recognition Letters*, 1987.

[17] R. Nevatia and K. Babu. Linear feature extraction and description. *Computer Graphics and Image Processing*, 13(3):257–269, 1980.

[18] F. Porikli, J. Shao, and H. Maehara. Extracting roads from aerial images using feature based classifiers. In *IAPR Conf. on Machine Vision Applications*, 2005.

[19] C. Poullis, S. You, and U. Neumann. A vision-based system for automatic detection and extraction of road networks. In *WACV*, pages 1–8, 2008.

[20] K. Price. Road grid extraction and verification. In *International Society for Photogrammetry and Remote Sensing*, pages 101–106, 1999.

[21] J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 22(8):888–905, 2000.

[22] R. Stoica, X. Descombes, and J. Zerubia. A gibbs point process for road extraction from remotely sensed images. *IJCV*, 57(2):121–136, 2004.

[23] C. Urmson, J. Anhalt, H. Bae, J. Bagnell, C. Baker, R. Bittner, T. Brown, M. Clark, M. Darms, D. Demitrish, J. Dolan, D. Duggins, D. Ferguson, T. Galatali, C. Geyer, M. Gittleman, S. Harbaugh, M. Hebert, T. Howard, S. Kolski, M. Likhachev, B. Litkouhi, A. Kelly, M. McNaughton, N. Miller, J. Nickolaou, K. Peterson, B. Pilnick, R. Rajkumar, P. Rybski, V. Sadekar, B. Salesky, Y.-W. Seo, S. Singh, J. Snider, J. Struble, A. Stentz, M. Taylor, W. Whittaker, Z. Wolkowicki, W. Zhang, and J. Ziglar. Autonomous driving in urban environments: Boss and the urban challenge. *Journal of Field Robotics*, 25(8):425–466, 2008.

[24] V. Vazirani. *Approximation Algorithms*. Springer-Verlag New York, 2004.

[25] P. Viola and M. Jones. Robust real-time object detection. In *IJCV*, 2001.

[26] J. Youn and J. Bethel. Adaptive snakes for urban road extraction. In *ISPRS Congress*, 2004.