

AUTOMATIC EXTRACTION AND REGULARIZATION OF BUILDING OUTLINES FROM AIRBORNE LIDAR POINT CLOUDS

Bastian Albers ^a, Martin Kada ^b, Andreas Wichmann ^b

^aInstitute for Geoinformatics and Remote Sensing (IGF), University of Osnabrück,
Barbarastraße 22b, 49076 Osnabrück, Germany
bastian.albers@uni-osnabrueck.de

^bInstitute of Geodesy and Geoinformation Science (IGG), Technische Universität Berlin,
Straße des 17. Juni 135, 10623 Berlin, Germany
{martin.kada, andreas.wichmann}@tu-berlin.de

Commission III, WG III/4

KEY WORDS: Buildings, Outlines, Algorithm, LiDAR, Extraction, Regularization

ABSTRACT:

Building outlines are needed for various applications like urban planning, 3D city modelling and updating cadaster. Their automatic reconstruction, e.g. from airborne laser scanning data, as regularized shapes is therefore of high relevance. Today's airborne laser scanning technology can produce dense 3D point clouds with high accuracy, which makes it an eligible data source to reconstruct 2D building outlines or even 3D building models. In this paper, we propose an automatic building outline extraction and regularization method that implements a trade-off between enforcing strict shape restriction and allowing flexible angles using an energy minimization approach. The proposed procedure can be summarized for each building as follows: (1) an initial building outline is created from a given set of building points with the alpha shape algorithm; (2) a Hough transform is used to determine the main directions of the building and to extract line segments which are oriented accordingly; (3) the alpha shape boundary points are then repositioned to both follow these segments, but also to respect their original location, favoring long line segments and certain angles. The energy function that guides this trade-off is evaluated with the Viterbi algorithm.

1. INTRODUCTION

Building outlines provide substantial information for the urban environment and are therefore needed to map urban variation and change. Typical applications are urban planning, city modelling and disaster management. Manually extracting building outlines or other urban features to keep data sets up to date is time and cost consuming.

Airborne images and laser scanning have been a major data source for building outline extraction. Automating methods, achieving more accurate results, faster processing of large data sets and transferability to other regions of interest are of main interest in ongoing research. Today's airborne laser scanning technology can produce dense 3D point clouds with high accuracy, which makes it an eligible data source to reconstruct 2D building outlines or even 3D building models.

Building extraction from LIDAR point clouds can be divided into four parts, i.e. (1) classify the point cloud to separate building points from ground, tree, and other urban feature points, (2) isolate individual buildings, (3) determine approximate hulls of boundary points, and (4) generalize/regularize building outlines (Kim and Shan, 2011). Regularization of building outlines is needed because initial building outlines from LIDAR point clouds are noisy and jagged. Depending on the requirements of the application, a generalization is often used to eliminate small details.

This paper focuses on the regularization of extracted building points from a LIDAR point cloud. The regularization is driven by an energy minimization evaluated by the Viterbi algorithm (Viterbi, 1967). The proposed approach enables a reconstruction

of building outlines with more than one main orientation and different angles by utilizing different methods to generate input for the energy function.

Section 2 gives an overview of related work in the field of regularizing building outlines mainly from airborne laser scanning data but also with examples from airborne images. Section 3 introduces the principles and methods used for the proposed approach in detail. Section 4 shows the formulated energy terms in detail. Section 5 examines experimental results of the algorithm tested on the ISPRS benchmark data set from Toronto. Conclusions and an outlook are given in Sections 6.

2. RELATED WORK

A good introduction to previous research in the field of regularization of building outlines is given by Jwa et al. (2008). A comparative analysis of four representative methods including Douglas-Peucker's algorithm, Local Minimum Description Length, Feature Based Model Verification and Rule-based Rectification is conducted. They also propose a geometric regularization based on Minimum Description Length (MDL) called Geometric MDL (GMDL). This method tries to find the optimal building polyline by adding a global directional constraint.

Huang and Sester (2011) utilize a hybrid method to extract and reconstruct building footprints. They first segment the 3D point cloud with a 3D Hough transform and then use the acquired information, i.e., roof heights and ridges as additional parameters for the statistical reconstruction of the building footprint with Reversible Jump Markov Chain Monte Carlo (RJCMC). The

proposed method only considers simple rectangular shapes and therefore has to be adapted for shapes that are more complex.

Jarzabek-Rychard (2012) applies Random Sample Consensus (RANSAC) to detect straight lines in a height image derived from a LIDAR point cloud. Regularization is then performed by merging close parallel line segments and adjusting angles according to a mean direction calculated from the longest line segments. Rectangularity and parallelism are used as hard constraints and therefore different orientations or angles are not considered.

Similar to the proposed method in this paper Fazan and Dal Poz (2013) present a building roof extraction based on snakes and dynamic programming, but performed on airborne images. An energy function is used to optimize the building outline and to refine the results. A drawback of the proposed method is that the weighting functions favor right angles and therefore only work for buildings with simple rectangular shapes.

Another approach based on energy minimization is presented by He et al. (2014). The alpha shape algorithm is used to delineate the initial building boundary, which is then simplified with an adapted Douglas-Peucker algorithm called Vertex-driven Douglas-Peucker, which utilizes energy minimization and focuses on the complexity of the resulting polygon. The regularization is then divided and tested in an explicit and implicit reconstruction that either focus on robustness and accuracy or completeness and topological correctness.

An exhaustive overview of current research regarding building extraction from airborne laser scanning data is also provided by Tomljenovic et al. (2015).

3. PROPOSED APPROACH

An overview of the proposed approach to extract and regularize building outlines is summarized in Figure 1. As this papers primary focus lies on the regularization by energy minimization, the first two steps of the proposed workflow (classification of point cloud and building point extraction) are only mentioned but not laid out in detail.

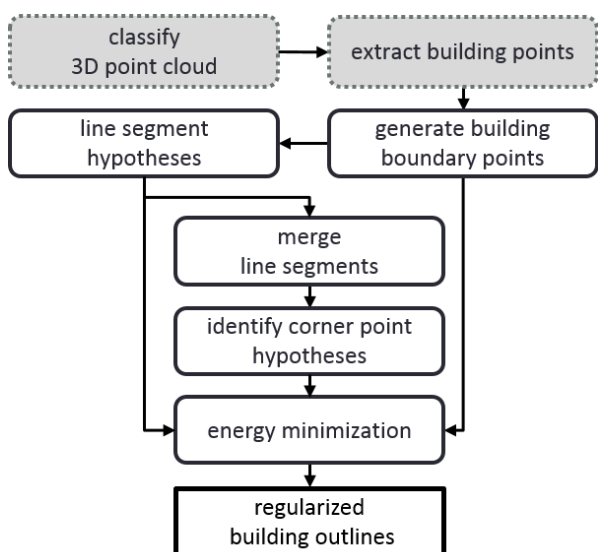


Figure 1. Workflow for regularizing building footprints.

For the regularization of building outlines, the alpha shape boundary points are considered as observations in a Markov Chain model. All extracted and delineated information (e.g. Hough transform line segments and corner point hypotheses) is then used as input for computing the transition probabilities with an energy formulation covering geometric properties suited for building outlines. The energy is then evaluated with the Viterbi algorithm to propose the optimal states for a regular building outline.

3.1 Building boundary points

The first step in creating an outline for a specific building starts with the generation of an approximate hull to extract all bounding points. The alpha shape algorithm (Edelsbrunner et al., 1983) is used for the given building points and is known to produce reliable building boundary points while preserving small details (Shahzad and Zhu, 2015, Dorminger and Pfeifer, 2008). Alpha shapes are a generalization of the convex hull of a point set but have the advantages of being able to be used for both convex and concave shapes and extracting polygons with interior and exterior boundaries (Shen et al., 2011). The alpha value can be adjusted to adapt the algorithm to different point cloud densities. As we are only interested in the boundary points, all remaining points are disregarded for further processing.

3.2 Line segment hypotheses

To add a first hypothesis about the approximate orientation of the regularized building outline we use the well-known Hough transform to detect line segments that represent the main directions of the building (Guercke and Sester, 2011, Duda and Hart, 1972). The main idea behind Hough transform is to perform a line detection in a parameter space, also called Hough space. In this parameter space, a line is no longer described as a line but by its parametric representation, in this case the normal parameterization as shown in this equation:

$$d = x_i \cos \alpha + y_i \sin \alpha \quad (1)$$

where d : distance from the origin
 α : direction of the line normal, restricted to $[0, 180]$
 x_i, y_i : point coordinates

The Hough space is defined by the variables α and d . Each point is transformed into a sinusoidal curve in the Hough space. The curves of collinear points intersect in the Hough space in one point. To find these intersections the Hough space is discretised according to a predefined angle and distance resolution. The resolution of each interval is a compromise of computation speed and accuracy. The resulting raster is also called Hough accumulator. Each grid cell, also called bin, counts how many curves are passing through. Bins with a high count correspond to a high confidence of a detected line segment. By thresholding the results, we can assure that bins with a small count are rejected and only bins with a favoured count are used for further processing.

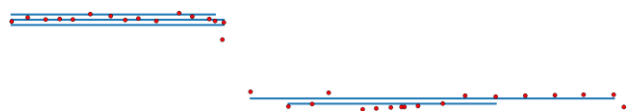


Figure 2. Detailed view of alpha shape boundary points (red) and detected line segments by adapted Hough transform (blue).

For our purposes, the Hough transform is slightly adapted as we are only interested in line segments aligned to the dominant

directions of a building. First, the Hough transform is used to delineate line segments. The longest line segment is then used to define the first dominant direction. Then all other aligned line segments ($angle \in \{45^\circ, 90^\circ, 135^\circ, 180^\circ\}$) are searched for and all boundary points in close proximity ($< 0.1m$) are removed. The Hough transform is then repeated with the reduced point list until no more dominant orientations are found. Figure 2 shows an excerpt of the Hough transform applied to the boundary points of an alpha shape. Additionally, a minimum length threshold of 1.5 m is used to reject small line segments that would disturb further processing.

3.3 Merging line segments

As shown in the previous chapter, the Hough transform can detect more than one line segment for a given set of points. An iterative line segment grouping based on parallelism and proximity followed by merging grouped line segments is used to reduce a group to one or more average line segments from which we can later compute intersection points.

First, the longest line segment l_{long} is added to L_{merge} and removed from the set of all detected line segments. Then each line segment in the set of remaining line segments L_{remain} is checked for parallelism to l_{long} followed by computing the perpendicular distance. If the distance is smaller than a given threshold, it is added to the set of line segments to be merged L_{merge} and removed from L_{remain} . When all line segments are processed, line segments in L_{merge} are further examined to see if they can be combined. For this, an average line is computed based on all given line segments in L_{merge} weighted by their length to function as a base line. Then a line tracing is started with the longest line segment. If other line segments overlap or are closer than 2 m a union is performed and the minimum and maximum extent of the longest line segment is changed accordingly. The computed extents are then projected on the average line to create the combined line segments. Figure 3 shows an example of four parallel line segments that are merged to one average line segment where a short gap is filled. One advantage of using all line segments in a group is that building objects like porches or balconies with small disparities to the building outline are all assigned to one base line. This information can later be used by the energy evaluation to align parallel or collinear line segments.



Figure 3. Line segments from Hough transform (blue) are merged to one or more collinear line segments (red).

After merging, the overall process continues by emptying L_{merge} and selecting again the longest line segment from L_{remain} and searching for close line segments. The process stops when L_{remain} is empty.

To know which line segments are adjacent to each other and for faster processing in subsequent steps, the resulting line segments are sorted. This is done by starting with an endpoint of the longest line segment and searching for the closest start point of any other line segment. The endpoint of the closest line segment is then used for the next search and so on.

3.4 Corner point hypotheses

Next to boundary points and line segment hypotheses, possible corner points of the building outline add important information for reconstruction. On the one hand hypothetical corner points

give information where a building wall probably ends and on the other hand give information where the dominant direction of a building changes and therefore hard angle constraints in further processing can be lowered.

Intersecting each consecutive merged line segment from the previous step results in a set of corner point hypotheses for further processing. However, it is possible that two consecutive line segments are parallel and therefore never share an intersection point. This usually happens when the Hough transform fails to detect a line segment because there were not enough boundary points to support the line segment or detected line segments were too short and were therefore rejected by given thresholds.

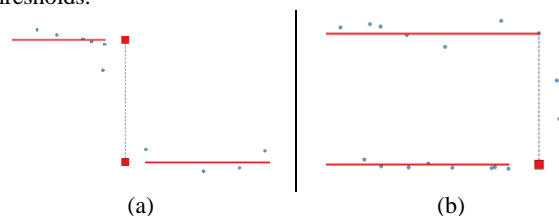


Figure 4. Special cases for adding corner points (red) for (a) Z-shape and (b) U-Shape with missing line segment (dotted).

In two cases, supplementary corner points are added when a line segment is probably missing between two detected parallel line segments: (1) the line segments form a Z-shape or (2) a U-Shape (Figure 4). For the first case, the extent from each line segment to the dropped foot of the perpendicular from the other line segment is computed. At the midpoint of each extent, a possible corner point is added. Figure 4 (a) illustrates how the first case overcomes a region of few alpha shape boundary points that lead to a missing line segment. For the second case, only the extent from the shorter line segment to the dropped foot of the perpendicular from the longer line segment is used to add one hypothetical corner point. Figure 4 (b) illustrates this case.

3.5 Energy evaluation

The Viterbi algorithm is a dynamic programming approach to compute the most likely sequence of observations given a specific Markov Chain (Viterbi, 1967, Forney, 1973). It is widely used for many applications like speech recognition, bio-informatics or digital communication. To apply the Viterbi algorithm the Markov model is formulated as follows:

- boundary points of the computed alpha shape are the given sequence of observations $X = x_1, x_2, \dots, x_n$
- we search for the most probable corresponding sequence of hidden states $Z = z_1, z_2, \dots, z_n$
- transition probabilities are defined as a transition matrix A_{ij} , where a_{ij} is the probability of moving from state i to state j .

For each observation a state grid is created, that relates to several potential candidates for each building outline point. To discretize the search space for all candidate hypotheses, the grid dimension is limited to the input data. Each observation is assigned to one merged line segment and its corresponding Hough line segments. After assignment, the states are computed with the following rules:

- the observation point is projected on each assigned line and a state grid with a specific resolution is created
- if a corner point is in close proximity of the observation point, an additional state grid is created around the corner point

The grid resolution is variable and depends on the point density of the point cloud. State points that superpose each other are removed to reduce complexity and time for the later energy computation. A clipped example of a resulting state grid is given in Figure 5. The four blue points represent alpha shape points, the blue lines are Hough line segments and the green line is the corresponding merged line, red points are the states. The bold red point is a corner point hypothesis and therefore an additional state grid is added to the far left alpha shape point.

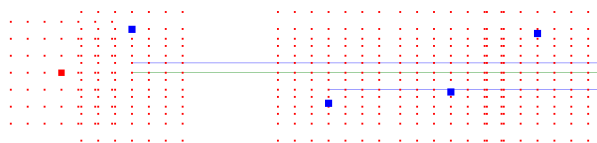


Figure 5. Example of a search space (red points) for four observations (blue points).

4. ENERGY FORMULATION

The proposed energy formulation has been developed with an emphasis on regularizing simple rectangular shapes, but also allowing angles to differ from 45° and 90°, when the underlying observation supports the change in main orientation. The proposed energy function is divided in three individual terms each with an emphasis on a specific geometric property:

$$E = \alpha E_{dist} + \beta E_{angle} + \gamma E_{length} \quad (2)$$

where α, β, γ : weight factors

The distance term E_{dist} keeps the solution close to the input observation; the angle term E_{angle} penalizes undesired angles; the length term E_{length} is used to prevent too many changes of direction and to keep the resulting polygon as simple as possible by preferring long line segments. The influence of each term is determined by weight factors α, β, γ . Depending on the input observations, the weight factors can be adjusted. In the presented experiments in Section 5 the weight factors are determined empirically but specific assumptions can be postulated beforehand. Both angle and length term are always weighted higher than the distance term, because the search space is relatively small and a too high weight on the distance term would fix the result to the original observation point. The angle and length term are mostly weighted equally as both terms are independent on the observation and each term should influence the regularization. Each term is explained in detail in the following sections.

4.1 Energy distances term

The distance term is used to increase the energy for estimated building outline points with a high distance from the observed point (boundary alpha shape point). A simple squared distance between both input points is computed:

$$E_{dist}(z, x) = \sum_{i=1}^n |z_i - x_i|^2 \quad (3)$$

where x_1, \dots, x_n : observed (alpha shape) points
 z_1, \dots, z_n : state grid points

4.2 Energy angle term

The angle term is used to weight occurring angles between each consecutive state and its corresponding line segment. By adding a specific weight, we favor certain angles but also allow unexpected angles in special cases. With this lenient function, we can avoid hard constraints:

$$E_{angle}(l) = \sum_{i=1}^m A(l_i, l_{(i+1) \bmod m}) \quad (4)$$

$$A(l_1, l_2) = \begin{cases} 0 & \text{for } \angle(l_1, l_2) \in \{180^\circ\} \\ 1.0 & \text{for } \angle(l_1, l_2) \in \{90^\circ\} \\ 1.25 & \text{for } \angle(l_1, l_2) \in \{45^\circ, 135^\circ\} \\ \tau & \text{otherwise} \end{cases} \quad (5)$$

where l_1, \dots, l_m : line segments

The weight factor τ should be adapted to the given data set e.g. in relation to the point density. If the value is set too low, changes in direction occur more frequent and disturb the overall regularization. In this study it is at least the doubled value of the weight given for the set of angles $\{45^\circ, 135^\circ\}$.

4.3 Energy length term

Including the length of a generated line segment to the energy leads to better results, because many short line segments are less likely in building outlines:

$$E_{length}(l) = \sum_{i=1}^m L(l_i) \quad (6)$$

$$L(l) = e^{\frac{1}{|l|}} \quad (7)$$

An exponential function is used for a smooth transition between the resulting energy for small and long line segments.

5. EXPERIMENTAL RESULTS

The regularization has been tested on the ISPRS benchmarking data set from Toronto with an average point density of about 6 points/m² (Rottensteiner et al., 2012). It consists of many large high-rise buildings that often cast shadows and occlude important building parts.

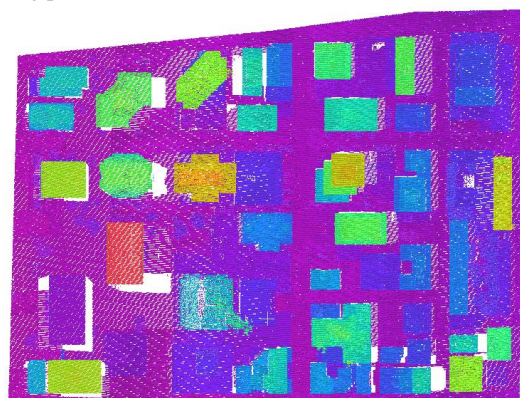


Figure 6. Overview of Toronto test area colored by height.

Figure 6 shows an overview of the test area with more than thirty buildings. Figure 7 shows the results of the building outline regularization. Figure 8 gives some examples of buildings with

more complex outlines outside the test area along with intermediate steps. Buildings (i) and (ii) show, that smaller indentations are removed but the overall shape remains intact. More complex shapes are shown with buildings (iii) and (iv). Overall, the proposed approach produces good results but can fail when the point density is too low or important building parts (e.g. smaller walls) are not represented in the data. Other important influences on the result are size and resolution of the search space. A bigger search space often leads to better results but also increases the processing time intensely. As He et al. (2014) note, the sensor scanning pattern also directly influences the results. The scanning pattern often leads to missing boundary points of roofs and therefore the line segment delineation with the Hough transform fails (for example building 6 in Figure 7).



Figure 7. Results of the building outline regularization for the Toronto test area.

Preliminary tests on a data set with a higher point density than the ISPRS data set showed an increased accuracy of the resulting outlines and an overall increased number of correctly extracted outlines.

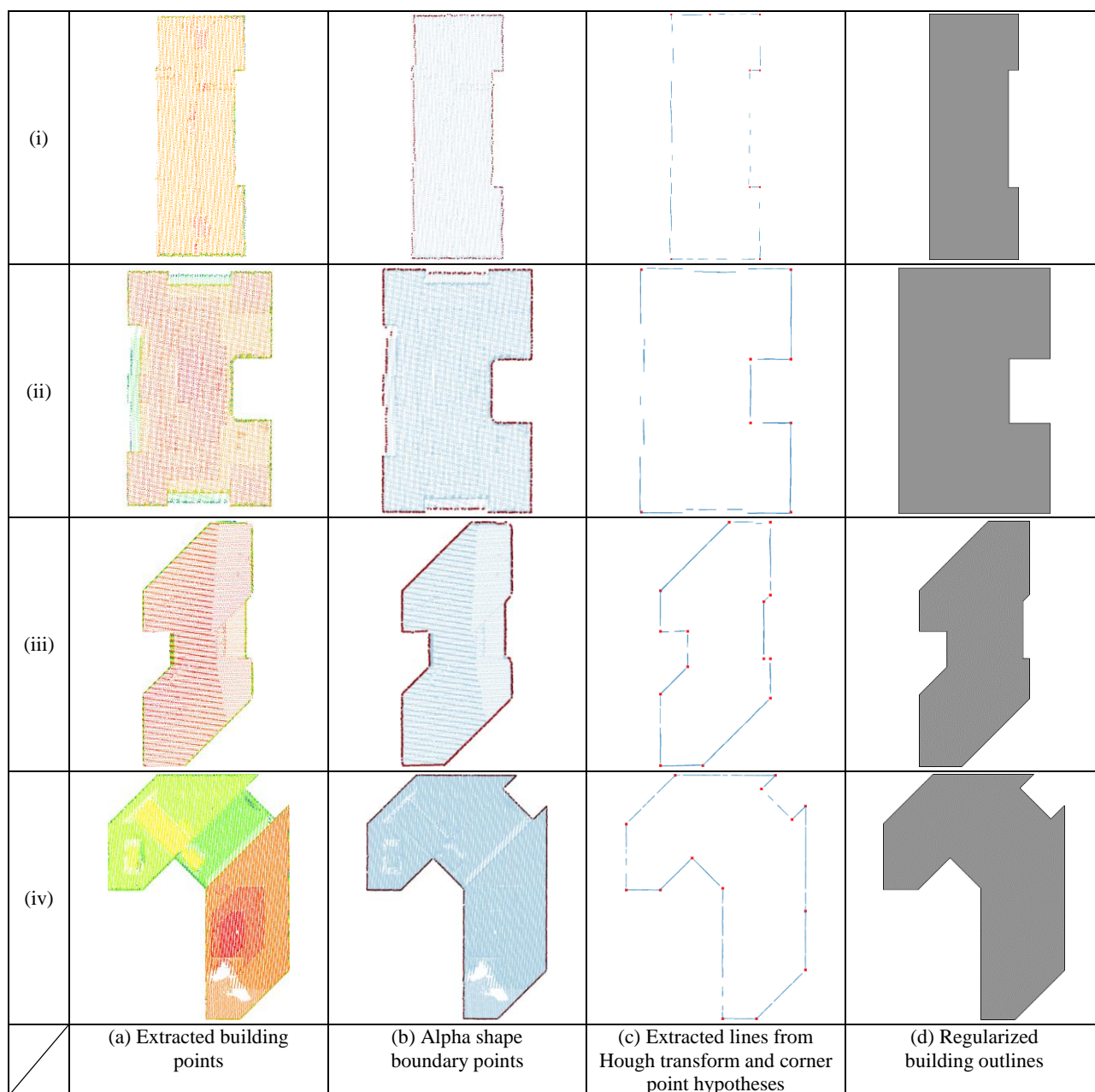


Figure 8. Results with intermediate steps for buildings from the Toronto data set.

6. CONCLUSION AND OUTLOOK

In this paper, an automated algorithm to regularize building outlines from LiDAR point clouds based on energy minimization has been shown. The energy function that guides this trade-off is evaluated with the Viterbi algorithm.

If the method can extract sufficient information from the boundary points of a building, the proposed approach shows good results for point clouds with a point density greater than 6 points/m². The overall shape of buildings is preserved while small anomalies are eliminated.

An advantage over other methods is that the presented approach allows more than one main orientation in buildings and can therefore represent shapes that are more complex.

We see the proposed approach as a first step towards a full processing pipeline to regularized 3D building roof outlines. To accomplish this task further testing on different datasets is planned and an evaluation with reference building outlines is needed. One aim is to include symmetries to the regularization process to consider recurring building shape parts. The next step would be to include several new energy terms to contemplate 3D features. Slope and topology of roof segments are, for example, suited for further examination.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the provision of the Downtown Toronto data set by Optech Inc., First Base Solutions Inc., GeoICT Lab at York University, and ISPRS WG III/4.

REFERENCES

Dorninger, P., Pfeifer, N., 2008. A comprehensive automated 3D approach for building extraction, reconstruction, and regularization from airborne laser scanning point clouds. *Sensors*, 8, 11, pp. 7323–7343.

Duda, R.O., Hart, P.E., 1972. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15, 1, pp. 11–15.

Edelsbrunner, H., Kirkpatrick, D., Seidel, R., 1983. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, 29, 4, pp. 551–559.

Fazan, A.J., Dal Poz, A.P., 2013. Rectilinear building roof contour extraction based on snakes and dynamic programming. *International Journal of Applied Earth Observation and Geoinformation*, 25, 1, pp. 1–10.

Forney, G.D., 1973. The Viterbi algorithm. *Proceedings of the IEEE*, 61, 3, pp. 268–278.

Guercke, R., Sester, M., 2011. Building footprint simplification based on Hough transform and least squares adjustment. In: *Proceedings of the 14th Workshop of the ICA Commission on Generalisation and Multiple Representation*, Paris.

He, Y., Zhang, C., Fraser, C.S., 2014. An energy minimization approach to automated extraction of regular building footprints from airborne LiDAR data. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3, September, pp. 65–72.

Huang, H., Sester, M., 2011. A hybrid approach to extraction and refinement of building footprints from airborne LiDAR data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 38, pp. 153–158.

Jarzabek-Rychard, M., 2012. Reconstruction of building outlines in dense urban areas based on LiDAR data and address points. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXIX, B3, pp. 121–126.

Jwa, Y., Sohn, G., Tao, V., Cho, W., 2008. An implicit geometric regularization of 3D building shape using airborne LiDAR data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXVII, B3a, pp. 69–76.

Kim, K., Shan, J., 2011. Building footprints extraction of dense residential areas from LiDAR data. *Proceedings of ASPRS Annual Conference*, pp. 193–198.

Lafarge, F., Descombes, X., Zerubia, J., Pierrot-Deseilligny, M., 2007. 3D city modeling based on hidden Markov model. *ICIP 2007, IEEE International Conference on Image Processing*, II, pp. 521–524.

Rottensteiner, F., Sohn, G., Jung, J., Gerke, M., Baillard, C., Benitez, S., Breitkopf, U., 2012. The ISPRS benchmark on urban object classification and 3D building reconstruction. *The International Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, I, 3, pp. 293–298.

Shahzad, M., Zhu, X.X., 2015. Reconstruction of building footprints using spaceborne TomoSAR point clouds. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3/W5, pp. 385–392.

Shen, W., Zhang, J., Yuan, F., 2011. A new algorithm of building boundary extraction based on LiDAR data. In: *19th International Conference on Geoinformatics*, Shanghai, pp. 1–4.

Tomljenovic, I., Höfle, B., Tiede, D., Blaschke, T., 2015. Building extraction from airborne laser scanning data: An analysis of the state of the art. *Remote Sensing*, 7, pp. 3826–3862.

Viterbi, A.J., 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13, 2, pp. 260–269.

Yang, B., Xu, W., Dong, Z., 2013. Automated extraction of building outlines from airborne laser scanning point clouds. *IEEE Geoscience and Remote Sensing Letters*, 10, 6, pp. 1399–1403.