ROOF PLANE EXTRACTION IN GRIDDED DIGITAL SURFACE MODELS

Ben Gorte

Optical and Laser Remote Sensing, Faculty of Aerospace Engineering,
TU Delft, the Netherlands
b.g.h.gorte@tudelft.nl

KEY WORDS: Algorithms, LIDAR, Modeling, City, Detection, Reconstruction

ABSTRACT:

With the rapid improvement of LIDAR systems regarding point density and accuracy in relation to the (application dependent) requirements, robustness, efficiency and automation of the modeling process are becoming more important than achieving the highest possible accuracy and modeling detail from the available LIDAR data. Therefore we opt for development of a 2D grid based LIDAR data analysis approach. An important step is detection and parameterization of planar surfaces (roof elements). The paper reviews four methods, based on analysis of gradients, principal components, least squares and hough transforms, respectively. It introduces a series of improvements to the standard usage of each of those methods and shows results from synthetic and real data.

1. INTRODUCTION

In nowadays information society a growing need for 3D information of the urban landscape is observed. Producers as well as consumers of geoinformation recognize the attractiveness of representing the urban environment by 3-dimensional models rather than 2D maps. Growing needs for efficient exploitation of the scarcely available urban space requires more careful and detailed spatial modeling than 2D information allows for. Examples can be found in urban planning and architecture, and in simulation and modeling of noise and pollution caused by traffic and industry. An interesting, new and fully 3-dimensional application is modeling the urban climate, with all its micro-climate phenomena. But also in consumer-oriented applications, such as real-estate advertising, portable and in-car navigation systems, and even Google Earth, 3D is becoming common practice.

With the increasing availability of airborne LIDAR data, full automation of 3D city modeling from those data has become a challenging and relevant research topic. In the algorithms to be applied a choice has to be made whether to regard the data as a 3D point cloud or as a 2D (often termed 2.5D) surface model, in which the elevation z is a function of the planimetric location (x,y). Furthermore, in both cases the data can be regarded as vector points with explicit coordinates, or be discretized in a regular grid with implicit raster coordinates. These choices lead to voxel representations in 3D or to pixel representations in 2D, where the elevations are stored in the pixel values.

Traditionally the goal of 3D modeling has been to obtain the best possible detail given the available data. For this reason vector representations are often considered superior over gridded ones, despite the advantages of the latter concerning efficiency and convenience. With the rapid improvement of LIDAR systems regarding point density and accuracy, however, the data are no longer the limiting factor, but robustness, efficiency and automation of data analysis have become more important instead. Therefore we opted for development of a 2D grid based approach for 3D modeling. An important step in the approach is detection and parameterization of planar surfaces (roof elements).

Irrespective of the chosen input data structure, the first phase of 3D building reconstruction can be described as a segmentation problem. It is necessary to subdivide the input point set into subsets corresponding to objects of interest, where objects are for example (depending on the required amount of detail) entire buildings of elements of building roofs. Subsequently (or simultaneously) the geometric characteristics (the shapes) of the objects are reconstructed on the basis of the coordinates of the participating points.

A choice to be made is whether the segmentation should be complete, meaning that every point is assigned to an object, or whether we are only looking for buildings, whereby points on vegetation and cars, or even on the ground, may be discarded as ‘not of interest’. In the latter case the entire issue is largely covered by the capability to detect planar regions above a certain height in a normalized DSM (which is a DSM where the terrain has $z=0$). Points below that height as well as those not belonging to planes are discarded. There are issues remaining, for example caused by non-planar roof elements, tall trees over buildings, roof gardens and trucks. Moreover, vertical building elements (walls) are not present in LIDAR point sets, but they have to be inferred from the data by recognizing surface discontinuities.

The remainder of the paper concerns detection, delineation and parameterization of planes. It reviews a number of methods in Section 2, with the purpose of combining these into a novel local Hough transform in section 3. Section 4 will show experimental results on synthetic and on real data, followed by conclusions and an outlook in Section 5.

2. PLANE DETECTION

Generally, plane detection can be performed using global or local methods. Global plane detection looks ‘at once’ at the entire data set or at a large subset, which is perhaps bounded by algorithm capacity or by prior knowledge concerning the maximum extends of a plane [Oude Elberink and Vosselman, 2006]. As a result, planes are entirely detected ‘at once’ as well. A popular global plane detection method is the Hough transform described below (Section 2.1). Local methods, at the other hand, attempt to decide for each point, or small group of nearby points, whether it might be part of a planar surface, and if so, what would be the surface parameters (Section 2.2). The decision is based on a neighborhood of the point or point group.

2.1 Hough Transforms

'Hough transform' is the collective name of a class of algorithms for detecting parameterized shapes in 2D or 3D data sets. The most popular example is detection of thin lines in two-dimensional binary images. The object pixels are supposed to belong to those lines, but there are also 'noise' object pixels, whereas the lines have gaps. The problem is to find the parameters of the lines, and the solution starts by parameterizing the set of image lines passing through an object pixel with coordinates (x, y) as (for example):

$$y = ax + b \tag{1}$$

In a parameter space with axes a and b this collection corresponds to a line

$$b = -xa + y \tag{2}$$

Therefore, a point in image space corresponds to a line in the parameter space. For a set of collinear image points the corresponding lines in parameter space intersect at a single point (a, b) representing the image line $y=ax + b$ that passes though all image points. Thus the problem of finding collinear points in image space is reduced to finding intersections of lines in parameter space (Figure 1). These intersections are easily found by discretizing the parameter space into a 2D accumulator array: an image where lines are constructed one by one, by adding the value 1 to all cells of the line. When all lines are done, each cell value represents the number of lines passing through this cell, and the location (a, b) of the cell denotes an image line passing through that many image points.

This principle is easily extended to finding planes

$$z = ax + by + c \tag{3}$$

in a set of 3D points (x,y,z), such as a LIDAR point cloud. Each 'image' point now corresponds to a plane

$$c = -xa - yb + z \tag{4}$$

in a (a,b,c) parameter space, and a point (a,b,c) where N planes intersect corresponds to a plane in image space containing N image points.

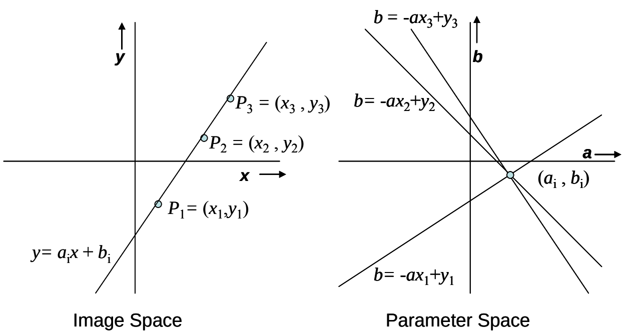


Figure 1. Hough transform for lines in 2D

Note: The above parameter spaces become unbounded when having vertical lines, resp. planes in the data. This is not a major problem in case of airborne LIDAR.

2.2 Local Plane Detection

Local planed detection methods analyze a small neighborhood of points at a time to decide about co-planarity of those points. Neighborhoods can be defined in two or in three dimensions. In 2D, a point near the edge on the roof of a high building and a point near to the wall on the ground may be in the same neighborhood, whereas in the 3D case they would not be.

It can be based, for example, on analysis of gradients (Section 2.2.1), least squares adjustment (2.2.2) or principal component analysis (2.2.3). At a later stage adjacent candidate points with similar surface parameter values are combined into larger planes, for example by region growing [Rottensteiner and Briese, 2001] (see Section 2.2.4), The performance of segmentation largely depends on the results of plane parameter estimation, which is the motivation of studying these closely.

The methods considered below can be described in terms of estimating parameters a, b and c of equation (3).

2.2.1 Gradient analysis

In a vector approach, plane parameters can be derived for each triangle after a Delaunay triangulation by using a voting mechanism similar to the one in Hough transforms, in order to construct larger segments from adjacent triangles with similar parameter estimates [Lohani and Singh, 2007]. [Gorte, 2002] presented a TIN-based region merging algorithm.

In a grid based approach, plane parameters a and b are derived straightforwardly as the image gradients in column and row direction, respectively, while taking the spatial resolution of the dataset into account. The value of the Laplacian, as estimated by a 3x3 filter in the same window, provides a measure for the planarity of the 3x3 neighborhood.

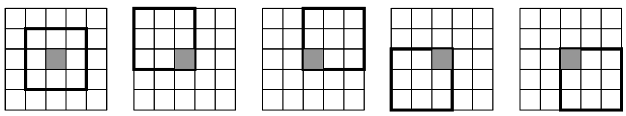


Figure 3. 3x3 subwindows in a 5x5 window

The main disadvantage of using gradients is that they are computed by subtracting neighboring z-values. Especially when point spacing (or grid spacing) is small compared to the measurement noise, gradients are getting quite noisy. In a grid-approach, the use of larger kernels reduces noise, but creates a wider zone near to the edges of planes where the results are unreliable.

Adaptive Gradient Filtering

An interesting way to reduce this edge effect is to consider different subwindows within the window around a pixel under consideration, as in Figure 2. This will be called adaptive gradient filtering. It is inspired by the Nagao type of edge-preserving smoothing [Nagao an Matsuyama, 1979], where the central pixel is assigned the average value of the subwindow with the smallest variance. Now, we assign the gradient from the subwindow with the smallest value for the laplacian. The same idea can be applied in a 9 x 9 window, using 5x5 subwindows. Figure 4 shows two of five cases.

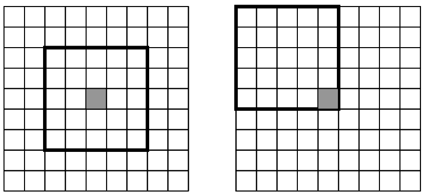


Figure 4.Two 5x5 subwindows in a 9x9 window

2.2.2 Least squares adjustment

A straightforward way of obtaining a plane through a number of non-collinear points is to perform a least squares estimation of the plane parameters. The RMS error provides a measure of the quality of the estimates, which is favorable since the estimates are severely contaminated if the plane does not fit well, as it happens near the edges of a plane, or in case of outliers in the points.

The method is easily implemented in a grid and extended in the same Nagao fashion as in the gradient method above, which we will call adaptive least squares filtering. window sizes are possible, for example using 7x7 subwindows in a 13x13 window, or 9x9 in 17x17. In each case the parameter estimates are obtained from the subwindow with the smallest RMS error.

2.2.3 Principal component analysis

When regarding a number ($N \geq 3$) of (x,y,z) points as a collection of simultaneous observations of variables x, y and z it is allowed to compute the 3x3 variance-covariance matrix C of these observations. Co-planarity of the points is signaled by the smallest eigenvalue of this matrix being (close to) zero, whereas the other two eigenvalues are significantly different from zero [Guru et al 2004]. The smallest eigenvalue can be used as a measure of co-planarity of the points. The eigenvectors belonging to the other two eigenvalues are orthogonal to each other and to the normal vector of the plane, and provide the plane parameter estimates.

Also now the extension to the distinction between different subwindows within a window under consideration is easily made. This can be called adaptive principal component filtering. Plane parameters are taken from the subwindow where the smallest eigenvalues is smallest (!).

It should be noted that the principal component method is equivalent to the least squares method of Section 2.2.2. Also computationally there is no clear advantage for either method.

2.2.4 Image segmentation

After plane parameters and co-planarity measures have been obtained by one of the local methods of Sections 2.2.1 tot 2.2.3, groups of adjacent points with similar parameters should be grouped in segments, corresponding to planar objects. It should be noted that all three plane parameters should be considered; using only the gradients a and b is not sufficient, for example, when two flat roofs with different heights (and different c values) are adjacent.

Popular image segmentation methods such as region growing and region merging rely on thresholds to determine whether adjacent pixels or regions are similar enough to be combined into a single segment. The performance of segmentation largely depends on the input, i.e. on the results of plane parameter estimation.

3. LOCAL HOUGH TRANSFORM

Hough transform, being a global method, is particularly suitable for detecting (and estimating parameters of) large structures, such as long lines in 2d or 3d, or big planes in 3d, that are sparsely represented by points in noisy images and point clouds. Its parameter estimation is quite insensitive to outliers. At the downside there is an element of chance in the detection because a parameter space resolution has to be chosen.

Moreover, the method does not consider adjacency of points being assigned to a plane (in the 3d case), Therefore a rather large threshold has to be set for the minimum plane size (i.e. the value in the accumulator array), or otherwise many arbitrary planes are generated from points that happen to be coplanar, but are spread all over the scene. Consequently, planes that are smaller than this threshold will not be detected. Sometimes this is solved by using prior knowledge, for example from 2D map data, to constrain the process to the interior of a single building ground plan, or even to a rectangle that is obtained by further subdividing a ground plan [Vosselman and Dijkman, 2001].

Another problem of Hough transforms in 3D is the computational cost. Having millions of points, belonging to hundreds of planes in a LIDAR data set, millions of planes need to be constructed in the parameter space. This space needs to have a resolution that allows hundreds of local maxima at the plane intersections to be represented and detected accurately.

Also here it helps to use prior knowledge to pre-segment the data, but a more general way out is to reduce the dimensionality of the problem. [Rabbani and van den Heuvel] for example managed to bring down the dimensionality of 5D cylinder parameter estimation by splitting the process into a 2D, followed by a 3D stage.

Combining these observations with the results of Section 2 inspired development of a local grid based Hough transform. Its purpose is to find the plane that passes through as many points as possible in a k x k window, including the point represented by the central pixel. The points are expressed in a local (x,y,z) coordinate system having the origin at the central pixel. The x and y coordinates of the other points are given by the row and column positions within the window, taking the spatial resolution into account, and the z value of each pixel is obtained by subtracting the central value from the pixel value.

The fact that (0,0,0) has to be part of the plane reduces the number of parameters from 3 to 2, since only planes

$$z = ax + by \tag{5}$$

need to be considered. For each of the k x k - 1 remaining points a line is constructed in an (a,b)-accumulator by:

$$b = z/y - ax/y \tag{6}$$

The maximum value in the accumulator determines the number of points, beside the central point, belonging to a single plane, whereas the position of this maximum in the accumulator gives an estimate for the corresponding plane parameters *a* and *b*. These are valid in the original coordinate system as well. Again, the remaining parameter *c* can be computed using equation (3). However, provided that sufficiently many points participate, a better estimate of all three parameters is obtained by a least squares fit of a plane through these points. Here, the absence of outliers in the set of participating points is of great benefit – the resulting RMS errors are expected to be much smaller than those obtained in Section 2.2.3.

4. EXPERIMENTS

The methods to estimate plane parameters from gridded DSMs described in section 3 are applied to three data sets:

1. a synthetic DSM of a simple house
2. a synthetic DSM of the same house with added noise
3. a FLI-MAP 400 dataset of an urban scene

The synthetic house was generated by sampling a point cloud from an “ideal” grid model with 10 cm resolution. The point cloud was then rotated over an angle of 28 degrees, and converted back to a 10 cm grid. Values of grid cells without points were interpolated form their neighbors by using a local average filter. The resulting DSM is shown in Figure 5.

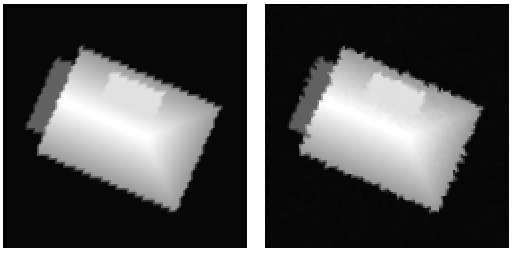
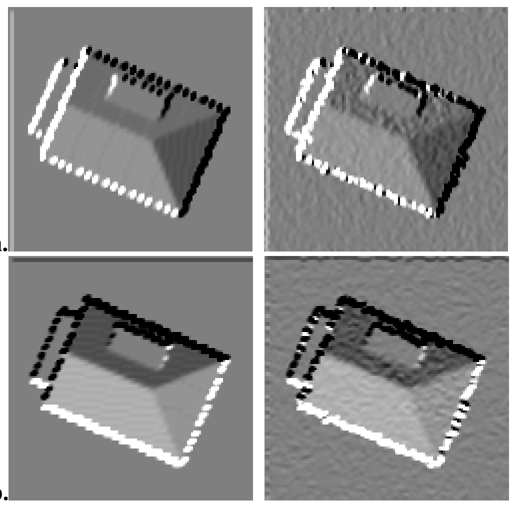


Figure 5.Synthetic DSM of a house with and without noise

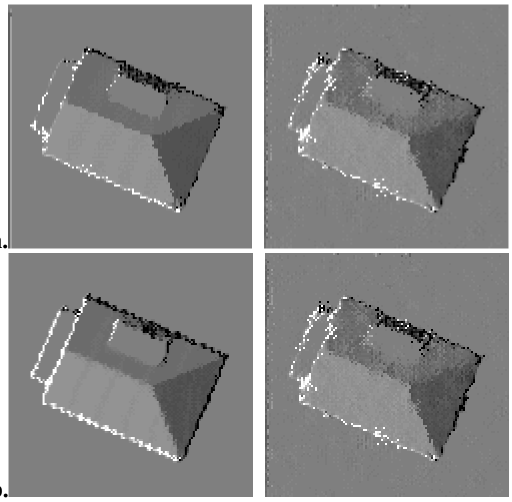
Figures 6 and 7 illustrate the gradient methods described in Section 2.2.1. Standard 3x3 image gradients in x and y direction, corresponding to estimates of the *a* and *b* plane parameters are shown (Figure 6), as well as adaptive gradient filters (only for the parameter *a*, Figure 7). It is clearly visible that the effect at the edges is drastically reduced in the latter case, and also noise is reduced significantly.



a.
b.

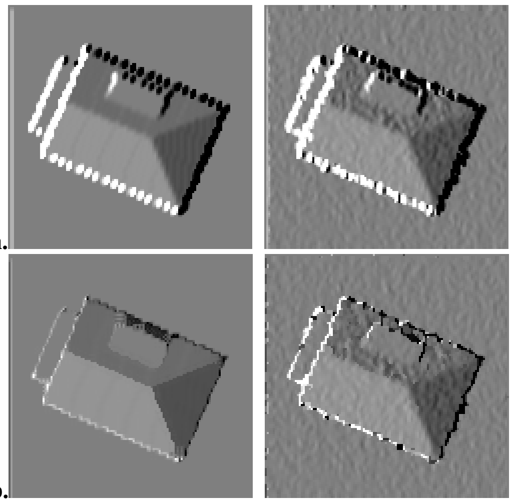
Figure 6. Gradient filtering.
a: image gradient in x direction without and with noise,
b: image gradient in y direction

The results of least squares methods is shown in figures 8 and 9. Figure 8 shows the estimates of the plane parameter *a* obtained by least squares filtering in a fixed 5x5 window and in an adaptive filtering in a 9x9 window with 5x5 subwindows, both for ideal and noisy images. Also the RMS errors of both situations are shown (Figure 9).



a.
b.

Figure 7. Adaptive gradient filtering.
a: 5x5 with 3x3 subwindows, without and with noise
b: 9x9 with 5x5 subwindows



a.
b.

Figure 8.. Least squares filtering in ideal and noisy data
a: in 5x5 window, b: adaptive 5x5 in 9x9 window

The next experiment concerns local Hough transforms where the *a* and *b* plane parameters are estimated directly from the positions of the maxima in the accumulator (only parameter *a* is shown). We show the results of different window sizes, again for ideal and noisy input data (Figure 10). It appears possible to apply large window sizes, resulting in effective noise reduction (outlier removal) without loosing spatial detail.

The final experiment on synthetic data concerns local Hough transform with simultaneous least squares plane parameter estimation. See Figure 11, where all parameter estimates are illustrated, as well as the RMS of least squares estimation. A 21x21 window size was used. It appears that the parameter estimates do not differ much from those directly obtained from local Hough transform with the same size and without least squares estimation (Figure 10d.). It should be noted that the RMS of the least squares estimates is very small compared to those in Figure 9, where the same mapping from RMS values to gray levels has been used.

Last but not least, a result from real data is shown in Figures 13.

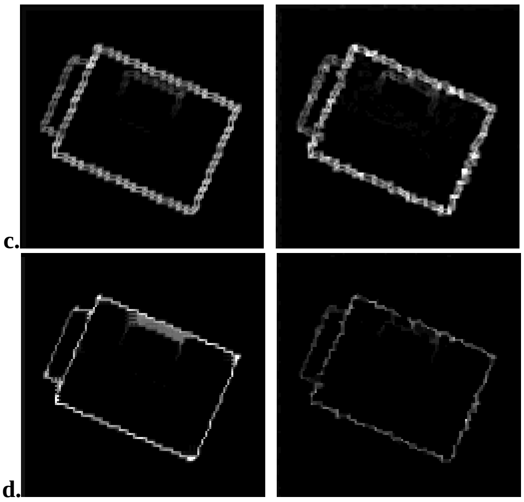
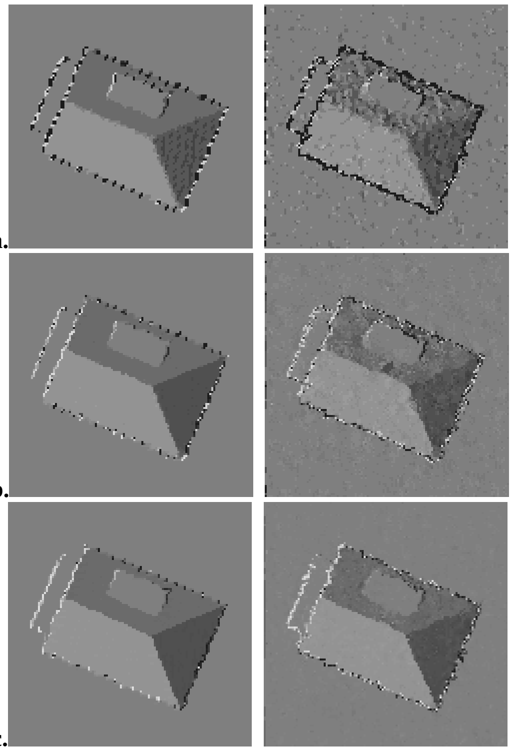

c.
d.

Figure 9.. Least squares filtering.
c: RMS of image 8a, d: RMS of image 8b.


a.
b.
c.

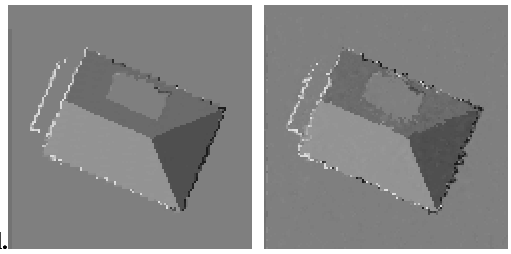

d.

Figure 10. Local Hough transform in ideal and noisy data
a: in 3 x 3 window, b: 7 x 7 c: 13 x 13, d: 21 x 21

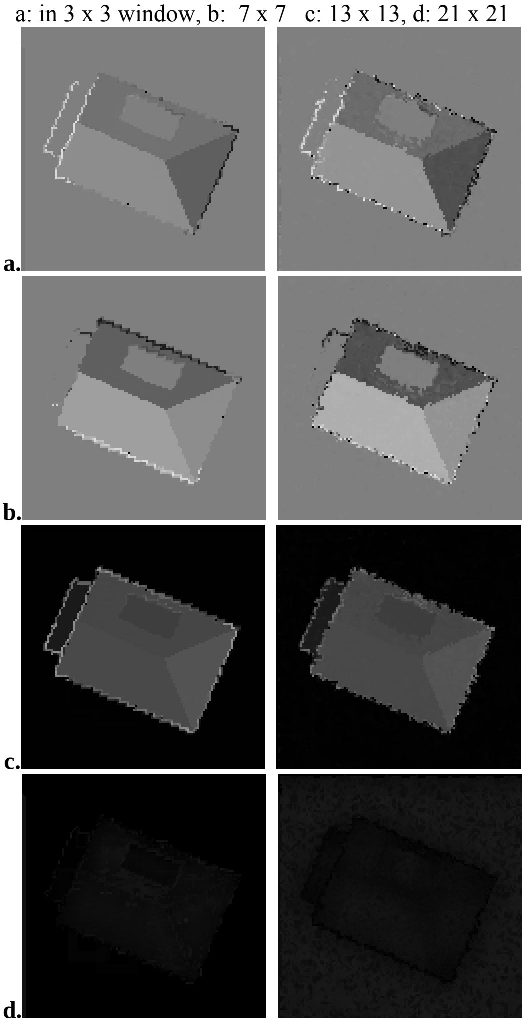

a.
b.
c.
d.

Figure 11. Local Hough transform with least squares plane parameter estimation of *a, b* and *c* in 21 x 21 windows.
d. shows the RMS of the least squares fit.

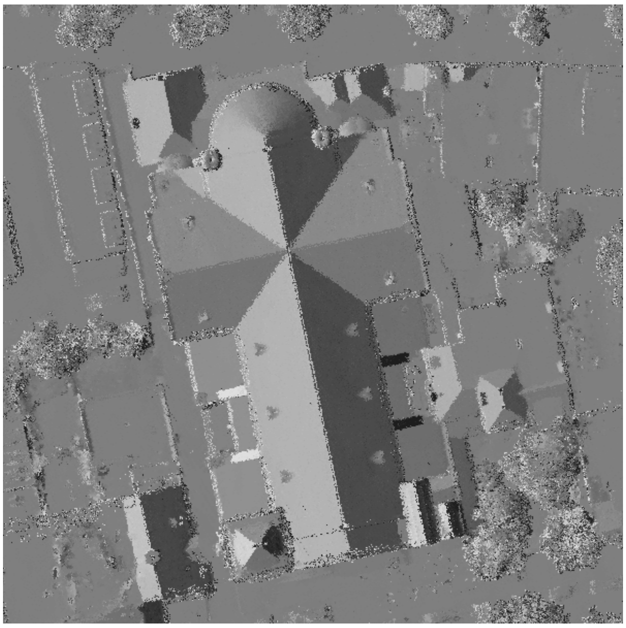


Figure 12: Plane parameter *a* estimated from FLI-MAP 400 data in Rotterdam (NL) using a 21x21 local Hough transform.

5. CONCLUSIONS AND OUTLOOK

The paper describes theoretical considerations leading to the development of a local Hough transform method for detecting planes in a gridded surface DSM, while estimating the plane parameters. The method was expected to deal effectively with noise and to behave well near the edges of plane, which is confirmed by experiments on synthetic data with and without noise, and appears to apply to real high-density LIDAR data in a complex urban scene as well.

The next step in the development of robust, fully automatic 3d city model generation will be the delineation of roof planes, both at the intersections of neighboring planes as well as at surface discontinuities where walls have to be reconstructed.

Acknowledgement: The FLI-MAP 400 data were collected by Fugro for the municipality of Rotterdam, and were made available for research purposes.

6. REFERENCES

[Guru et al. 2004] D.S. Guru, B.H. Shekar, P. Nagabhushan, A simple and robust line detection algorithm based on small eigenvalue analysis, *Pattern Recognition Letters* 25 (2004) 1–13

[Vosselman and Dijkman, 2001] Vosselman, G. and Dijkman, S. (2001). 3d building model reconstruction from point clouds and ground plans. *IAPRS 34 part 3/W 4* Commission III., pp.37- 44 .

[Oude Elberink and Vosselman, 2006] Sander Oude Elberink and George Vosselman, Adding the third dimension to a topographic data base using airborne laser scanning data, *IAPRS XXXVII* , commission III, Bonn, 2006.

[Rabbani and van den Heuvel, 2005] Tahir Rabbani and Frank van den Heuvel, Efficient Hough transform for automatic detection of cylinders in point clouds, ISPRS WG III/3, III/4, V/3 Workshop "Laser scanning 2005", Enschede, NL.

[Gorte, 2002] Ben Gorte, Segmentation of TIN-Structured Surface Models. In: *Proceedings Joint International Symposium on Geospatial Theory, Processing and Applications*, Ottawa, Canada.

[Lohani and Singh, 2007] Bharat Lohani and Rajneesh Singh, Development of a Hough transform based algorithm for extraction of buildings from actual and simulated LiDAR data, *Mapworld Forum 2007*, Hyderabad, 22-25 January 2007

[Rottensteiner and Briese, 2001] Rottensteiner, F., Briese, Ch., 2002. A new method for building extraction in urban areas from high resolution LIDAR data. ISPRS. Photogrammet. Comput. Vision, Graz, Austria, 9–13 September, pp. A-295 ff.

[Nagao and Matsuyama, 1979], M. Nagao and T. Matsuyama, Edge preserving smoothing, *Computer graphics and image processing* 9(4), April 1979