Fusion of Feature- and Area-Based Information for Urban Buildings Modeling from Aerial Imagery*

Lukas Zebedin[1], Joachim Bauer[1], Konrad Karner[1], and Horst Bischof[2]

[1] Microsoft Photogrammetry
irstname.lastname@microsoft.com
[2] Graz University of Technology
bischof@icg.tugraz.at

Abstract. Accurate and realistic building models of urban environments are increasingly important for applications, like virtual tourism or city planning. Initiatives like Virtual Earth or Google Earth are aiming at offering virtual models of all major cities world wide. The prohibitively high costs of manual generation of such models explain the need for an automatic workflow.

This paper proposes an algorithm for fully automatic building reconstruction from aerial images. Sparse line features delineating height discontinuities and dense depth data providing the roof surface are combined in an innovative manner with a global optimization algorithm based on Graph Cuts. The fusion process exploits the advantages of both information sources and thus yields superior reconstruction results compared to the indiviual sources. The nature of the algorithm also allows to elegantly generate image driven levels of detail of the geometry.

The algorithm is applied to a number of real world data sets encompassing thousands of buildings. The results are analyzed in detail and extensively evaluated using ground truth data.

1 Introduction

Algorithms for the semi- or fully automatic generation of realistic 3D models of urban environments from aerial images are subject of research for many years. Such models were needed for urban planning purposes or for virtual tourist guides. Since the advent of web-based interactive applications like Virtual Earth and Google Earth and with the adoption of 3D content for mashups the demand for realistic models has significantly increased. The goal is to obtain realistic and detailed 3D models for entire cities.

This poses several requirements for the algorithm: First, it should not require any manual interaction because this would induce high costs. This restriction also dissuades the use of cadastral maps as they vary in accuracy, are not readily available everywhere and require careful registration towards the aerial data. Additionally such a dependency increases the cost at large scale deployment. Second, the algorithm should be flexible enough to generate accurate models for common urban roof structures without limiting itself to one specific type, like gabled roofs or rectangular outlines for example. This

* This work has been supported by the FFG project APAFA (813397) under the FIT-IT program.

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part IV, LNCS 5305, pp. 873–886, 2008.

also includes the requirement to be able to deal with complex compositions of roof shapes if those happen to be adjacent. Third, the algorithm should have a certain degree of efficiency as it is targeted at thousands of cities with millions of buildings in total. Last, the algorithm should be robust: the visual appearance should degrade gracefully under the presence of noise or bad input data quality.

In the following a survey and assessment of existing algorithms is given, which fail to meet one or more of the above mentioned requirements.

Among the early approaches are feature based modelling methods ([1,2,3,4,5]) which show very good results for suburban areas. The drawback of those methods is their reliance on sparse line features to describe the complete geometry of the building. The fusion of those sparse features is very fragile as there is no way to obtain the globally most consistent model.

The possibility of using additional data (cadastral maps and other GIS data in most cases) to help in the reconstruction task is apparent and already addressed in many publications ([6,7,8]). Such external data, however, is considered manual intervention in our work and thus not used.

A different group of algorithms concentrates on the analysis of dense altimetry data obtained from laser scans or dense stereo matching ([9,10]). Such segmentation approaches based solely on height information, however, are prone to failure if buildings are surrounded by trees and require a constrained model to overcome the smoothness of the data at height discontinuities. Guhno and Downman ([11]) combined the elevation data from a LIDAR scan with satellite imagery using rectilinear line cues. Their approach was, however, limited to determining the outline of a building. In our work we develop this approach further and embed it into a framework which overcomes the problems described above.

In [12] we have proposed a workflow to automatically derive the input data used in this paper. The typical aerial images used in the workflow have 80% along-strip overlap and 60% across-strip overlap. This highly redundant data is utilized in this paper. Similar approaches have been proposed by others ([13,14]), which demonstrate that it is possible to automatically derive a digital terrain model, digital elevation model, land use classification and orthographic image from aerial images. Figure 1 illustrates

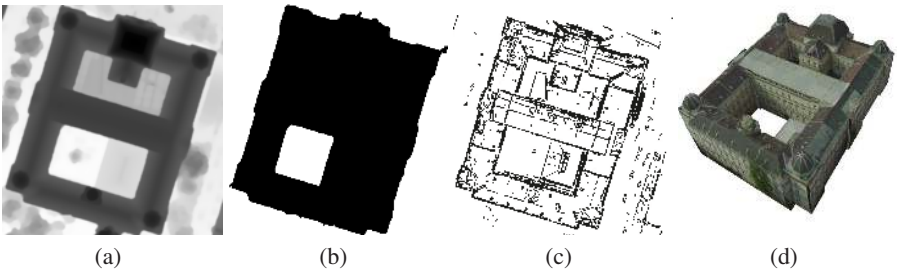


Fig. 1. These figures depict the data which is used for the reconstruction process: (a) height field, (b) building mask and (c) 3D line segments. Image (d) shows the obtained model by our algorithm.

the available data which is used for the reconstruction algorithm and also shows the result of the proposed algorithm.

Our proposed method does not need any manual intervention and uses only data derived from the original aerial imagery. It combines dense height data together with feature matching to overcome the problem of precise localization of height discontinuities. The nature of this fusion process separates discovery of geometric primitives from the generation of the building model in the spirit of the recover-and-select paradigm ([15]), thus lending robustness to the method as the global optimal configuration is chosen. The integration of the theory of instantaneous kinematics ([16]) allows to elegantly detect and estimate surfaces of revolution which describe a much broader family of roof shapes. A major feature of the proposed method is the possibility to generate various levels of geometric detail.

The rest of the paper is structured as follows: Chapter 2 gives a general overview of the method. In Chapter 3 we will describe the discovery of geometric primitives which are used to approximate the roof shape, whereas Chapter 4 discusses the building segmentation. Chapter 5 gives details about the fusion process which combines line features and dense image data. Results and experiments are outlined in Chapter 6. Finally, conclusions and further work are described in Chapter 7.

2 Overview of the Method

The workflow of the proposed method is outlined in Figure 2. Three types of information are necessary as input for the algorithm: Dense height data is generated by a dense image matching algorithm ([17]) (Figure 1a, represented as a height field) and gives a good estimate of the elevation, but suffers from oversmoothing at height discontinuities ([18]). Additionally a rough segmentation of the building is required (Figure1b) which could be directly deduced from the height data for example. The third component are sparse 3D line segments (Figure1c) which are obtained from line matching over multiple views ([1]).

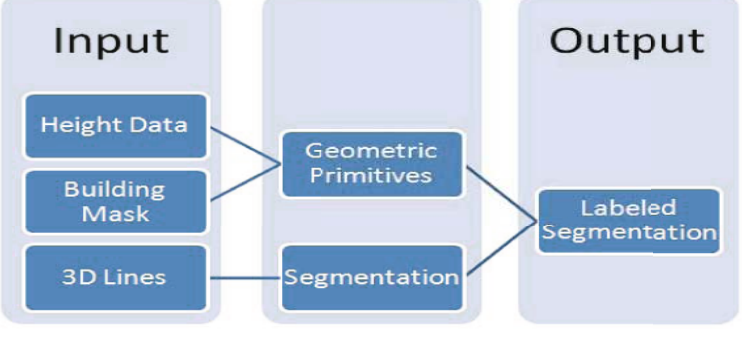


Fig. 2. Illustration of the single steps of the proposed method: height data and building mask are used to obtain a set of geometric primitives; In parallel the 3D lines are used to generate a segmentation of the building. Finally, a labeled segmentation is produced.

The building mask is combined with the dense height data, thus filtering out all 3D points which do not belong to the building. Afterwards the remaining points are grouped into geometric primitives. The geometric primitives are the basic building blocks for assembling the roof shape.

The 3D line segments are projected into the height field and used to obtain a line-based segmentation of the building. The 2D lines of the segmentation form polygons which are then assigned to one of the geometric primitives. Therefore, it is important that the 3D lines capture the location of the height discontinuities as each polygon is treated as one consistent entity which can be described by one geometric primitive. By extruding each of the 2D polygons to the assigned geometric primitive a 3D model of the building is generated.

Note that the algorithm presented in this paper makes no assumptions about the roof shape. Façades are modeled as vertical planes, because the oblique angle of the aerial images does not allow a precise reconstruction of any details.

3 Geometric Primitives

Geometric primitives form the basic building blocks which are used to describe the roof shape of a building. Currently two types of primitives, namely planes and surfaces of revolution, are used, but the method can be trivially extended to support other primitives. It is important to note, that the detection of geometric primitives is independent from the composition of the model. This means that an arbitrary amount of hypotheses can be collected and fed into later stages of the algorithm. As the order of discovery of the primitives is not important, weak and improbable hypotheses are also collected as they will be rejected later in the fusion step. If a primitive is missed, the algorithm selects another detected primitive instead which minimizes the incurred reconstruction error.

3.1 Planes

Efficiently detecting planes in point clouds for urban reconstruction is well studied and robust algorithms are readily available ([9]). Thanks to the independence of hypothesis discovery and model selection, a region growing process is sufficient in our workflow for the discovery of planes. Depending on the size of the building a number of random seed points are selected, for which the normal vector is estimated from the local neighbourhood. Starting from the seed points, neighbours are added which fit the initial plane estimate. This plane is regularly refined from the selected neighbours. Small regions are rejected to improve the efficiency of the optimization phase. Due to their frequency, close to horizontal planes are modified to make them exactly horizontal, the other oblique ones are left unchanged.

3.2 Surfaces of Revolution

Planar approximations of certain roof shapes (domes and spires for example) obtained from plane fitting algorithms, however, are not robust, visually displeasing and do not take the redundancy provided by the symmetrical shape into account. Therefore it is

necessary to be able to deal with other shapes as well and combine them seamlessly to obtain a realistic model of the building.

Surfaces of revolution are a natural description of domes and spires and can be robustly detected. Mathematically such surfaces can be described by a 3D curve which moves in space according to an Euclidean motion. Instantaneus kinematics gives a relationship ([19]) between that Euclidean motion parameters and the corresponding velocity vector field. Using that connection it is possible to estimate the parameters of the Euclidean motion in a least squares sense given the normal vectors of the resulting surface.

The equation

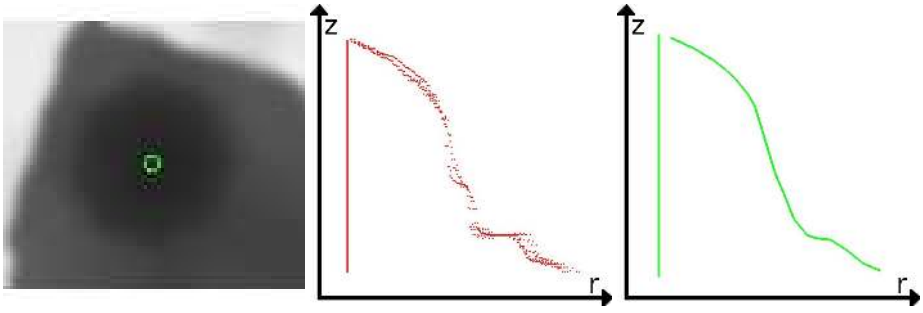
$$v(x) = \bar{c} + c \times x \tag{1}$$

describes a velocity vector field with a constant rotation and constant translation defined by the two vectors $c, \bar{c} \in \mathbb{R}^3$. If a curve sweeps along that vector field, the normal vectors of all points on the resulting surface have to be perpendicular to the velocity vector at the associated point. Thus

$$\begin{aligned} n(x)v(x) &= 0 \\ n(x)\left(\bar{c} + c \times x\right) &= 0 \end{aligned} \tag{2}$$

holds, where $n(x)$ gives the normal vector at point x. With equation (2) it is possible to estimate the motion parameters given at least six point and normal vector pairs $(x, n(x))$ lying on the same surface generated by such a sweeping curve. In the case of point clouds describing an urban scene the parameter can be constrained by requiring the rotation axis to be vertical. This already reduces the degrees of freedom to two (assuming that z is vertical) and makes the problem easily solvable:

$$\bar{c} = (0, x, y)^T \qquad c = (0, 0, 1)^T$$



(a) (b) (c)

Fig. 3. Illustrations how starting with the dense height data the 3D curve is derived which generates the dome if it rotates around a vertical axis. (a) Raw height field with the detected axis, (b) all inliers are projected into the halfplane formed by axis and a radial vector, (c) the moving average algorithm produces a smooth curve.

where $\bar{c}$ gives the position of the axis and c denotes the vertical rotation axis. The remaining two unknown parameters are estimated by transforming each 3D point with the estimated normal vector $(x, n(x))$ into a Hough space ([20]). Local maxima in the accumulation space indicate axes for surfaces of revolution. For each axis all inliers are computed and projected into the halfplane spanned by the rotation axis and an arbitrary additional radial vector. The redundancy of the symmetrical configuration can be exploited by a moving average algorithm in order to estimate a smooth curve which generates the surface containing the inliers. Figure 3 illustrates those steps with a point cloud describing the shape of a spire.

4 Segmentation

The goal of the segmentation is to represent the general building structure - not only a rectangular shape - as a set of 2D polygons.

The approach of Schmid and Zisserman ([21]) is used for the generation of the 3D line set that is then used for the segmentation of the building into 2D polygons. A 3D line segment must have observations in at least four images in order to be a valid hypothesis. This strategy ensures that the reliability and geometric accuracy of the reported 3D line segments is sufficiently high. The presence of outliers is tolerable since the purpose of the 3D lines is to provide a possible segmentation of the building. Any 3D line that does not describe a depth discontinuity can be considered as an unwanted outlier which will contribute to the segmentation, but will be eliminated in the fusion stage.

The matched 3D line segments are used to obtain a 2D segmentation of the building into polygons by appying an orthographic projection. The 2D lines cannot be used directly to segment the building, however, as the matching algorithm often yields many short line segments describing the same height discontinuity. A grouping mechanism merges those lines to obtain longer and more robust lines. A weighted orientation

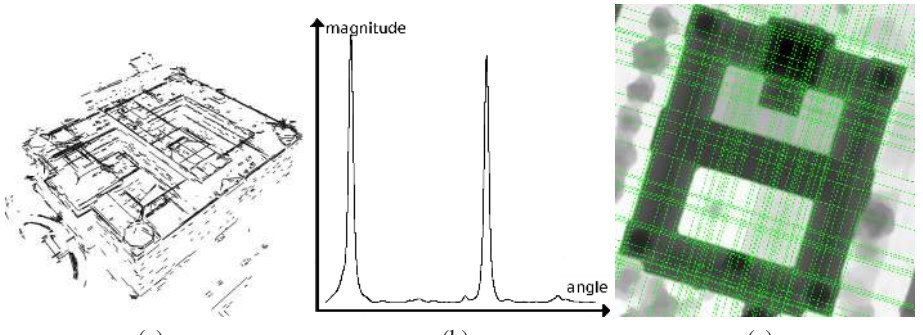


Fig. 4. Segmentation into polygons: (a) The matched 3D lines are projected into the $2\frac{1}{2}$D height field, (b) outliers are eliminated by a weighted orientation histogram which helps to detect principal directions of the building. (c) Along those directions lines are grouped, merged and extended to span the whole building.

histogram - the weights correspond to the length of each line - is created. The principal orientations are detected by finding local maxima in the histogram. Along those directions quasi parallel lines are grouped and merged thus refining their position.

Each grouped line is extended to span the whole building in order to simplify the segmentation process. The lines are splitting the area into a number of polygons. Each polygon is considered to be one consistent entity where the 3D points can be approximated by one geometric primitive.

Figure 4 illustrates this concept. The advantage of this approach is that no assumption or constraint of the shape, angles and connectivity of the building is necessary.

5 Information Fusion

Each polygon resulting from the segmentation is assigned to one geometric primitive (plane or surface of revolution, see Chapter 3). This labeling allows to create a piecewise planar reconstruction of the building - surfaces of rotation are approximated by a rotating polyline and therefore also yield piecewise planar surfaces in the polyhedral model.

The goal of the fusion step is to approximate the roof shape by the geometric primitives in order to fullfill an optimization criterion. In this paper we use the Graph Cuts algorithm with alpha-expansion moves ([22,23]), but other techniques like belief propagation are suited as well. The goal of this optimization is to select a geometric primitive for each polygon of the segmentation and to find an optimal trade-off between data fidelity and smoothness.

5.1 Graph Cuts Optimization

The Graph Cuts algorithm finds a very good approximation of the globally optimal solution for a broad range of tasks which can be stated as an energy minimization problem of the following form:

$$E(f) = \sum_{p \in P} D_p(f_p) + \lambda \cdot \sum_{\{p,q\} \in N} V_{p,q}(f_p, f_q) \tag{3}$$

where $V_{p,q}(f_p, f_q)$ is called the smoothness term for the connected nodes p and q which are labeled f_p and f_q and $D_p(f_p)$ is called the data term which measures a data fidelity obtained by assigning the label f_p to node p.

In our approach the segmentation induces a set P of polygons, where each polygon represent a node of the graph. The neighboorhood relationship is reflected by the set N, which contains pairs of adjacent polygons, ie. polygons sharing an edge. The set of labels used in the optimization process represent the geometric primitives (planes and surfaces of revolution):

$$L = \{\text{plane}_1, \text{plane}_2, ..., \text{surface-of-revolution}_1, \text{surface-of-revolution}_2, ...\} \tag{4}$$

Thus $f_p \in L$ reflects the label (current geometric primitve) assigned to node (polygon) $p \in P$.

The optimization using polygons is much faster than optimizing for each individual pixel because there are much fewer polygons than pixels. On the other hand it also exploits the redundancy of the height data because it is assumed that all pixels in one polygon belong to the same geometric primitive.

In our context the smoothness term measures the length of the border between two polygons and the data term measures the deviation between the observed surface (obtained from the dense image matching algorithm) and the fitted primitive. The following formulae are used to calculate those two terms:

$$D_p(f_p) = \sum_{x \in p} \left| height_{obs}(x) - height_{f_p}(x) \right| \tag{5}$$

$$V_{p,q}(f_p, f_q) = \begin{cases} length(border(p,q)) & \text{if} f_p \neq f_q \\ 0 & \text{if} f_p = f_q \end{cases} \tag{6}$$

where p and q denote two polygons and f_p is the current label of polygon p. The preset constant λ can be used to weight the two terms in the energy functional. The data term D_p calculates an approximation of the volume between the point cloud ($height_{obs}(x)$) and primitive f_p ($height_{f_p}(x)$) by sampling points x which lie within the polygon p. This sampling strategy allows to treat all geometric primitives similarly. because they are reduced to the incurred difference in volume and induced border to other polygons assigned to another geometric primitive. The smoothness term $V_{p,q}$ penalizes neighbouring polygons with different labels depending on their common border, thus favouring homogeneous regions.

The alpha-expansion move is used in order to efficiently optimize the labeling of all polygons with respect to all discovered primitives. The initial labeling can either be random or a labeling which minimizes only the data term for each individual polygon. After a few iterations (usually less than 5), the optimization converges and all 2D polygons can be extruded to the respective height of the assigned primitive to generate a polyhedral model of the building.

5.2 Levels of Detail

The second term in Equation (3) regularizes the problem and favors smooth solutions. Depending on the actual value of λ in Equation (3) different results are obtained. Higher values result in fewer and shorter borders at the cost of larger volumetric differences

Table 1. The impact of the smoothness parameter λ on the reconstructed model. The number of unique labels used after the Graph Cuts optimization iterations decreases as well as the number of triangles in the polygonal model. Δ Volume denotes the estimated difference in volume between the surface obtained by dense image matching and the reconstruced model (data term). The last column refers to the accumulated length of all borders in the final labeling (smoothness term).

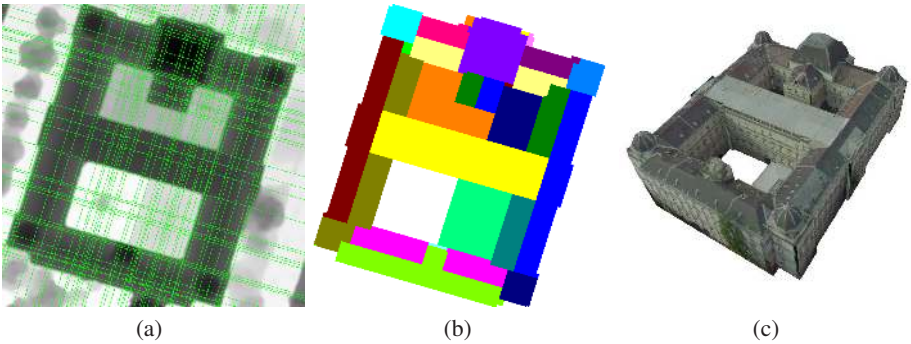
λ	#Labels	#Triangles	Δ Volume [m^3]	Border Length [m]
5	7	79	1210.79	710.4
10	6	69	1677.19	349.4
20	4	42	1699.31	337.0
100	3	33	2293.36	290.4

between observed height values and reconstructed models. This feature can be used to generate different models with varying smoothness, trading data fidelity for geometric simplificationa as smaller details of the building are omitted. An example of such a simplification is shown in Figure 6. The relevant numbers for that building are given in Table 1.

6 Experiments

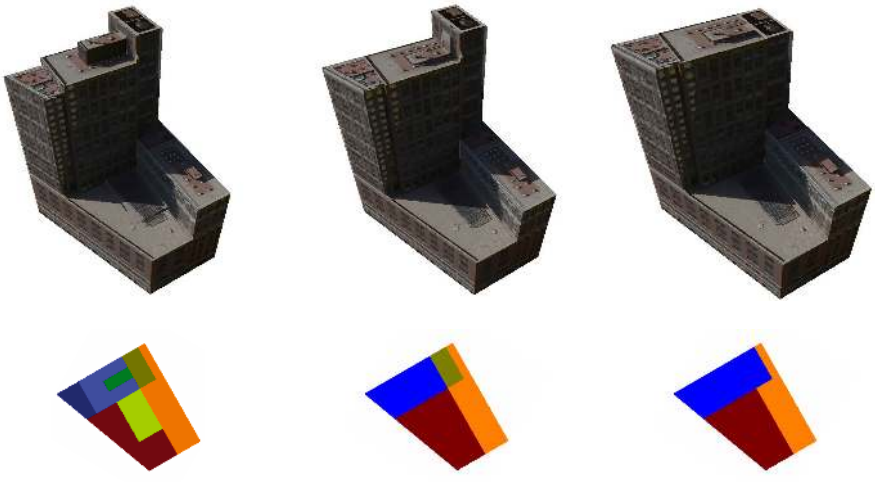
The first illustrative experiment was conducted on a test data set of a Graz. The ground sampling distance of the aerial imagery is 8cm. The examined building features four small cupolas at the corners. Additionally one façade is partially occluded by trees. Figure 5 shows the results of the reconstruction process. The texture of the façades is well aligned, implying that their orientation was accurately estimated by the 3D line matching. The domes are smoothly integrated into the otherwise planar reconstruction. Even the portion occluded by the tree has been straightened by the extension of the matched 3D lines.

The next example is taken from a data set of Manhattan, New York. This building shows that the reconstruction algorithm is not limited to façades perpendicular or parallel to each other. Figure 6 illustrates the effect of the smoothness term in the global optimization energy function. Various runs with different values for λ yield a reduced triangle count as the geometry is progressively simplified. Table 1 gives details about the solution for different values of λ. The Graph Cuts algorithm allows to find a globally optimal tradeoff between data fidelity and generalization. Those properties are expressed by the decreased length of borders and number of labels (which translate in general to fewer triangles) at the cost of an increase of the average difference between reconstructed and observed surface.



(a) (b) (c)

Fig. 5. The stages of the reconstruction are illustrated by means of the building of the Graz University of Technology: (a) Segmented height field, (b) labeled polygons after the Graph Cuts optimization, (c) screenshot of the reconstructed model ($\lambda = 5$)

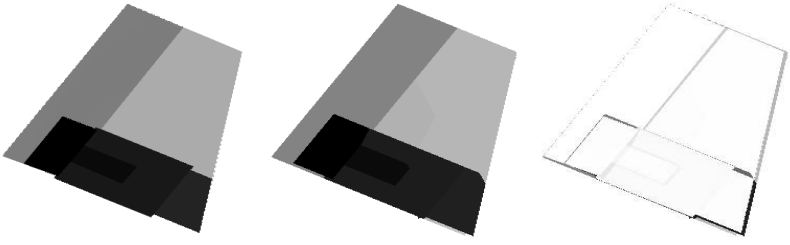


(a) 7 Primitives, $\lambda = 5$ (b) 4 Primitives, $\lambda = 20$ (c) 3 Primitives, $\lambda = 100$

Fig. 6. Levels of Detail: The same building was reconstructed with different values for λ. The number of geometric primitives used to approximate the shape of the roof is decreasing with higher values for λ. In the upper row a screenshot of the reconstruction is depicted, below are illustrations of the matching labeling obtained by the Graph Cuts optimization.

Apart from judging the visual appearance of the resulting models, we assess the quality of the reconstructed models by comparing them to a ground truth which was obtained manually from the same imagery. For this purpose we use a stereoscopic device to trace the roof lines in 3D. Those roof lines are connected to form polygons and then extruded to the ground level. Those manually reconstructed models are considered ground truth data in this paper. Using this procedure the whole data set from Manhattan (consisting of 1419 aerial images at 15cm ground sampling distance) was processed yielding 1973 buildings.

A comparison of manual and automatic reconstruction for one building is illustrated in Figure 7. Both building models are converted into a height field with a ground sampling distance of 15cm. This makes it easy to determine and illustrate their differences. Figure 8 gives a break down of the height differences as a cummulative probabilty distribution. Those graphs give the percentage of pixels where the height difference between manual and automatic reconstruction is lower than a certain threshold. Analysis of this chart shows that for the whole data set of Manhattan (1973 buildings) 67.51% of the pixels have a height difference smaller than 0.5m, 72.85% differ by less than 1m and 86.91% are within 2m. There are two main reasons for discrepancies of height values: On the one hand there are displacement errors of roof edges which lead to large height differences, depending on the height of the adjacent roof. On the other hand the human operator is able to recognize small superstructurial details on the roofs like elevator shafts and air conditioning units which cause height differences usually below 2m. Those small features are sometimes missed by the automatic reconstruction.



(a) (b) (c)

Fig. 7. Quality assessment with a manually generated ground truth: In (a) and (b) the height fields for the manually and automatically reconstructed building are shown, in (c) the height differences are shown. The largest difference in the placement of edges is about two pixels, which is about 30cm.

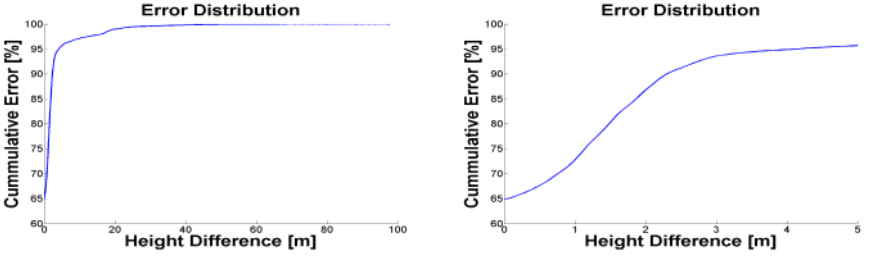


Fig. 8. The cummulative probabilty distribution of the height difference for manual and automatic reconstruction. The graph shows the error distribution for 1973 buildings from a data set of Manhattan, New York. The left image shows the graphs for height differences up to 100 meters; the right graph zooms on differences up to five meters.

Detailed views of typical results from the Manhattan data set are shown in Figure 9. The reconstruction of rectangular buildings is very successful, even though huge portions of their façades are occluded by trees. The integration of surfaces of revolution realistically models domes and spires (see 9b and 9d). It is important to note that for the purpose of visualization the surfaces of revolution are converted to triangle meshes by sampling them regularly (2m radially with 45 degrees of angular separation).

7 Conclusions and Future Work

In this paper we proposed a novell approach to reconstruct building models from aerial images by combining 3D line segments and dense image matching algorithms with a global optimization technique. The framework is able to use arbitrary basic geometric building blocks to describe the roof shape. The proposed surfaces of revolution elegantly describe domes and spires which are difficult to recover with an approach based on planes only. The combination of line based features and dense image matching

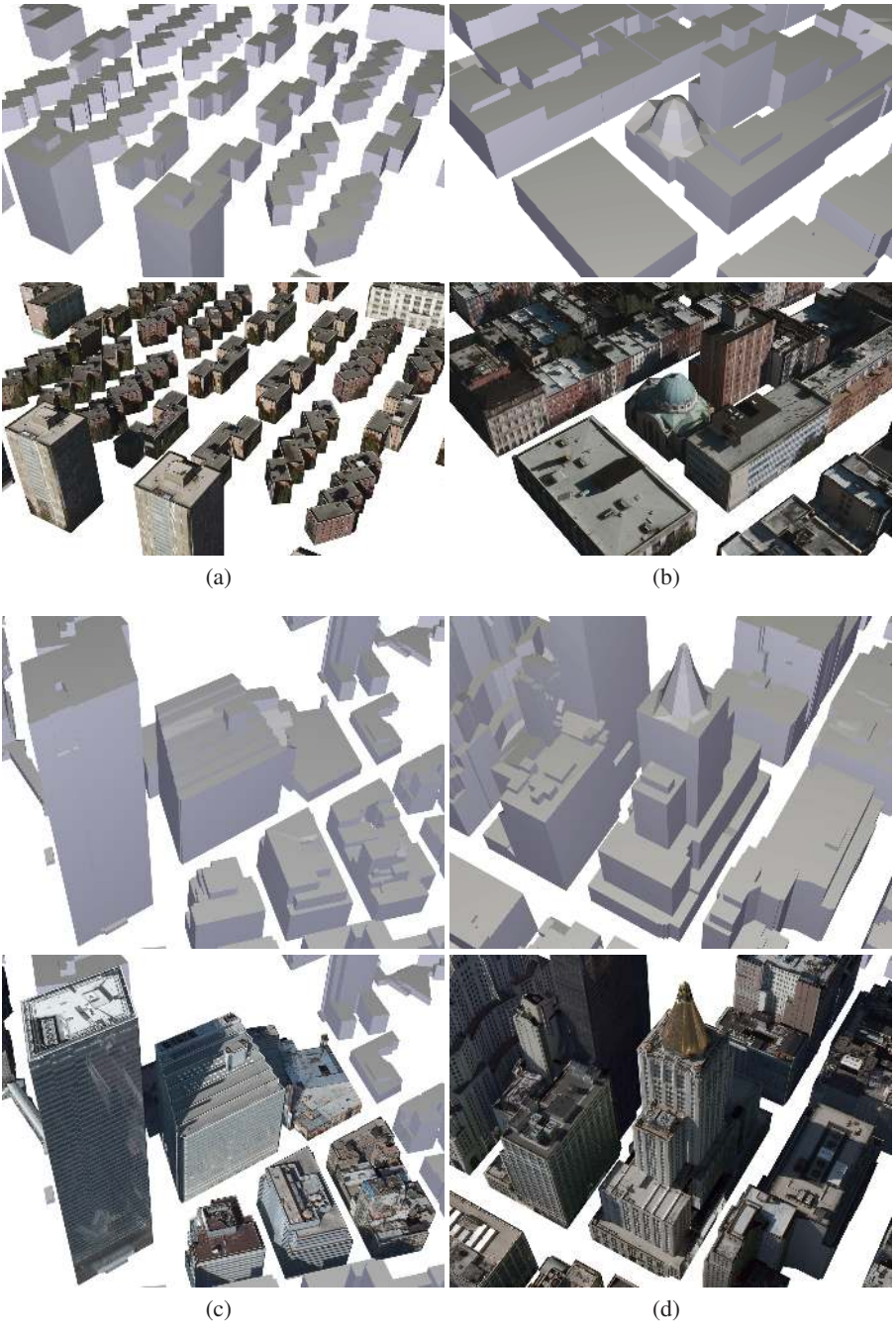


Fig. 9. Four detailed views of typical results for different types of buildings from the Manhattan data set: (a) rectangular buildings, (b) rectangular building with nicely integrated dome, (c) skyscrapers in downtown and (d) skyscraper with a spire

algorithms using a global optimization technique is very promising and is not restricted to the reconstruction of urban scenes from aerial imagery. Additionally it allows for the generation of different globally optimal levels of detail.

Future work will involve the investigation of other geometric primitives and methods to exploit symmetries encountered in common roof shapes like gabled roofs. Further research will be needed to evaluate the possibilities of this approach in other applications like streetside imagery.

References

1. Baillard, C., Zisserman, A.: Automatic Line Matching And 3D Reconstruction Of Buildings From Multiple Views. In: ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery, vol. 32, pp. 69–80 (1999)
2. Bignone, F., Henricsson, O., Fua, P., Stricker, M.A.: Automatic Extraction of Generic House Roofs from High Resolution Aerial Imagery. In: European Conference on Computer Vision, Berlin, Germany, pp. 85–96 (1996)
3. Fischer, A., Kolbe, T., Lang, F.: Integration of 2D and 3D Reasoning for Building Reconstruction using a Generic Hierarchical Model. In: Workshop on Semantic Modeling for the Acquisition of Topographic Information, Munich, Germany, pp. 101–119 (1999)
4. Taillandier, F., Deriche, R.: Automatic Buildings Reconstruction from Aerial Images: a Generic Bayesian Framework. In: Proceedings of the XXth ISPRS Congress, Istanbul, Turkey (2004)
5. Vosselman, G.: Building Reconstruction Using Planar Faces in Very High Density Height Data. In: ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery, Munich, vol. 32, pp. 87–92 (1999)
6. Baillard, C.: Production of DSM/DTM in Urban Areas: Role and Influence of 3D Vectors. In: ISPRS Congress, Instanbul, Turkey, vol. 35, p. 112 (2004)
7. Haala, N., Anders, K.H.: Fusion of 2D-GIS and Image Data for 3D Building Reconstruction. In: International Archives of Photogrammetry and Remote Sensing, vol. 31, pp. 289–290 (1996)
8. Suveg, I., Vosselman, G.: Reconstruction of 3D Building Models from Aerial Images and Maps. ISPRS Journal of Photogrammetry and Remote Sensing 58(3-4), 202–224 (2004)
9. Haala, N., Brenner, C.: Generation of 3D City Models from Airborne Laser Scanning Data. In: 3rd EARSEL Workshop on Lidar Remote Sensing on Land and Sea, Tallinn, Estonia, pp. 105–112 (1997)
10. Maas, H.G., Vosselman, G.: Two Algorithms for Extracting Building Models from Raw Laser Altimetry Data. In: ISPRS Journal of Photogrammetry and Remote Sensing, vol. 54, pp. 153–163 (1999)
11. Sohn, G., Dowman, I.: Data Fusion of High-Resolution Satellite Imagery and LIDAR Data for Automatic Building Extraction. ISPRS Journal of Photogrammetry and Remote Sensing 62(1), 43–63 (2007)
12. Zebedin, L., Klaus, A., Gruber-Geymayer, B., Karner, K.: Towards 3D Map Generation from Digital Aerial Images. ISPRS Journal of Photogrammetry and Remote Sensing 60(6), 413–427 (2006)
13. Chen, L.C., Teo, T.A., Shaoa, Y.C., Lai, Y.C., Rau, J.Y.: Fusion of Lidar Data and Optical Imagery for Building Modeling. In: International Archives of Photogrammetry and Remote Sensing, vol. 35(B4), pp. 732–737 (2004)
14. Hui, L.Y., Trinder, J., Kubik, K.: Automatic Building Extraction for 3D Terrain Reconstruction using Interpretation Techniques. In: ISPRS Workshop on High Resolution Mapping from Space, Hannover, Germany, p. 9 (2003)

15. Leonardis, A., Gupta, A., Bajcsy, R.: Segmentation of Range Images as the Search for Geometric Parametric Models. International Journal of Computer Vision 14(3), 253–277 (1995)
16. Pottmann, H., Leopoldseder, S., Hofer, M.: Registration without ICP, vol. 95, pp. 54–71 (2004)
17. Klaus, A., Sormann, M., Karner, K.: Segment-Based Stereo Matching Using Belief Propagation and a Self-Adapting Dissimilarity Measure. In: Proceedings of the 18th International Conference on Pattern Recognition, vol. 3, pp. 15–18. IEEE Computer Society Press, Washington (2006)
18. Scharstein, D., Szeliski, R.: A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. International Journal of Computer Vision 47, 7–42 (2002)
19. Pottmann, H., Leopoldseder, S., Hofer, M.: Simultaneous Registration of Multiple Views of a 3D Object. In: Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 34, Part 3A (2002)
20. Illingworth, J., Kittler, J.: A Survey of the Hough Transform. Computer Vision, Graphics and Image Processing 44(1) (1988)
21. Schmid, C., Zisserman, A.: Automatic Line Matching Across Views. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 666–671 (1997)
22. Boykov, Y., Veksler, O., Zabih, R.: Fast Approximate Energy Minimization Via Graph Cuts. In: International Conference on Computer Vision, Kerkyra, Corfu, vol. 1, pp. 377–384 (1999)
23. Kolmogorov, V., Zabih, R.: What Energy Functions Can Be Minimized Via Graph Cuts? In: European Conference on Computer Vision, Copenhagen, Denmark, vol. 3, pp. 65–81 (2002)