A Graph-based Approach for Automatic Building Extraction from Aerial LIDAR Data

Vivek Verma, Rakesh Kumar
Sarnoff Corporation
Princeton, NJ
vverma@sarnoff.com, rkumar@sarnoff.com

Stephen Hsu
Canesta
Sunnyvale, CA
shsu@canesta.com

Victor Santiago
PMTRASYS
Orlando, FL
victor.santiago@navy.mil

ABSTRACT

Realistic 3D city models are used in important applications such as flight simulators, war game simulations, security & surveillance, entertainment, etc. Many of these applications require that the model be as close as possible to the real world that it simulates, both in terms of geometry and texture. Recently, aerial LIDAR (Light Detection and Ranging) has gained popularity as a way to quickly collect 3D information about a site. LIDAR scanning of a site produces dense, unorganized points that require further processing to identify buildings, trees, and bare ground.

We present a fully automatic approach to creating 3D geometric models of the buildings and terrain from LIDAR data. Our objective is to create compact, watertight geometric models of buildings that fit as close as possible to the original LIDAR points using the minimum number of triangles, as opposed to a dense mesh. We propose to model the class of buildings that can be constructed by combining several simpler prismatic models. Such primitives are also amenable to automatic semantic interpretation as well as intuitive interactive editing.

We first segment the building roofs, trees, and terrain from the LIDAR points. We next fit local planar patches to the segmented roof points that are then grouped together to identify individual faces of the roof as polygons. By analyzing their proximity to each other, we construct a planar graph that represents the topology of the roof structure, including adjacency, symmetry, and orthogonality constraints between roof faces. Instances of simple prismatic models in the scene, such as hip roof configurations, can be identified by subgraph matching. The geometric parameters of these models are then refined based on the original point cloud. Using the above approach we can automatically model complex roofs by combining simpler prismatic objects.

ABOUT THE AUTHORS

Vivek Verma is a member of technical staff in the Vision Technologies Group at Sarnoff Corporation, Princeton, NJ. Before joining Sarnoff Corporation, he worked in the 3D graphics group at Autodesk, Inc. He received PhD (2001) and MS (1998) degrees from the Computer Science Department at the University of California, Santa Cruz and Master of Science (1994) degree from the Department of Mathematics at Indian Institute of Technology, Delhi, India. His research interests are in the areas of scientific visualization and computer graphics.

Stephen Hsu received the B.S. in electrical engineering from Caltech in 1982 and the Ph.D. in electrical engineering from MIT in 1988. Steve was an employee of Sarnoff from 1988 to 2005. The work presented here was done while Steve was at Sarnoff. He is currently a staff engineer at Canesta in Sunnyvale, CA. His research pursuits have included visual perception, image restoration, pattern recognition for image indexing and object recognition, low bit-rate video compression via mosaic-based prediction, algorithms for image registration and mosaic construction, and 3D pose and scene reconstruction.

Rakesh Kumar is technical director in the Vision and Robotics Laboratory at Sarnoff Corporation, Princeton, New Jersey. He is also an adjunct assistant professor at the Computer Science department at the University of Massachusetts at Amherst. Prior to joining Sarnoff, he was employed at IBM. He received his Ph.D. in Computer Science from the University of Massachusetts at Amherst in 1992. He received his MS in Electrical and Computer Engineering from the State University of New York at Buffalo in 1985 and his B.Tech. degree in Electrical Engineering from the Indian Institute of Technology, Kanpur, in 1983. His research interests are in computer vision, computer graphics and image processing.

Victor Santiago serves as a Lead Systems Engineer and Project Officer for the USMC Program Manager for Training Systems (PMTRASYS) Science and Technology Division. He earned an M.S. in Computer engineering from Widener University, Chester PA in 1998. At PMTRASYS, Victor has teamed with Sarnoff Corporation to conduct research in information visualization for MOUT training. Under the ONR-funded Real-Time Visualization research effort, Victor guided Sarnoff in demonstrating the effectiveness of Sarnoff's Video Flashlights system for military training. He is currently working to expand the capabilities of Sarnoff's Real-Time Visualization software system platform for future visualization research efforts. In addition, Victor is working on the rapid deployability of this platform, with the vision of making it a transitional asset for USMC training abroad. His research interests are visualization, modeling computer graphics and animation, and communication infrastructures.

A Graph-based Approach for Automatic Building Extraction from Aerial LIDAR Data

Vivek Verma, Rakesh Kumar
Sarnoff Corporation
Princeton, NJ
vverma@sarnoff.com, rkumar@sarnoff.com

Stephen Hsu
Canesta
Sunnyvale, CA
shsu@canesta.com

Victor Santiago
PMTRASYS
Orlando, FL
victor.santiago@navy.mil

INTRODUCTION

The past decade has witnessed the emergence of 3D graphics technology applied to various applications in commercial, military, and entertainment areas. This has been possible due to the advances in computing power, maturity of graphics hardware and algorithms, as well as improvements of sensors and devices such as digital cameras, displays, etc.

The military uses of computer graphics technology include applications such as flight simulators, situational awareness, simulation & training, etc. The PMTRASYS office in Orlando, FL is championing the use of computer graphics in creating next generation training systems for the Marines. The next generation of Marines will hone their skills by using these computer simulations. The benefits of these systems are:

1. Repeatability of an exercise
2. Non-life threatening if a mistake is made
3. Performance evaluation and progress tracking

Rapid generation of urban landscapes has been identified as one of the "science & technology long poles in the tent (Bailey and Armstrong, 2004). They have identified that it is important to rapidly generate valid urban databases for the Marines to train effectively and realistically for urban operations. Currently, it takes too long to create fully featured urban database. An urban database would contain realistic models of the terrain geometry (buildings and landscape), lighting effects, rubble, microclimate, population, etc. In this work our goal is to significantly speedup the geometric modeling of an urban landscape.

LIDAR (Light Detection and Ranging) is increasingly becoming the modality of choice to obtain 3D information for outdoor as well as indoor scenes. LIDAR data can be rapidly collected over an urban area that requires to be modeled. We present a system to automatically extract complex buildings and terrain from LIDAR data. This would pave the way for fast creation of geo-specific urban models for training as well as mission rehearsal.

Our goal is to automatically recover an accurate, complete, and compact geometric model of a 3D scene from a large collection of LIDAR point samples. The model should explain as much of the LIDAR data as possible and also interpolate over gaps in the data, commonly caused by occlusion or low reflectance. For realism, the model should obey real-world characteristics such as perpendicularity of features, sharpness of corners, and freedom from gaps between surfaces (water-tightness). For efficient rendering, this triangulated model should be as simplified as possible, e.g., a rectangular wall should be represented as two large triangles, not thousands of tiny mesh elements.

BACKGROUND

Training, mission planning, and mission execution for warfare in urban battlefields can be enhanced by 3D visualization of the environment and forces. Geometrically and visually accurate site models of buildings and urban environments – both indoors and outdoors – need to be constructed with short lead times. All conventional real-time 3D graphics pipelines are designed to render textured triangulated models, in which the shape and appearance of a scene are represented by a collection of 3D triangular facets, each bearing a digital image of that surface. Currently, constructing such models from images alone is a labor-intensive process involving skilled photogrammetrists and graphic artists. Some researchers have developed systems that combine novel user interfaces with some amount of automation (Lee 2002) but the method depends on the availability of accurate pose information about the cameras that took the photographs used in the modeling process.

The emergence of COTS/GOTS scanning laser range finders and position sensors would seem to neatly bypass the fundamental difficulties of reconstructing 3D shape and camera poses from 2D image collections. Yet, rapid and highly automated construction of accurate and compact models is an unfulfilled goal. This is due to challenges in computer vision/graphics algorithms, user interfaces, and system integration. Approaches to create realistic3D geometric models of urban areas range from completely manual methods using multiple photographs of the buildings (ReEALViz, Ulm 2003) to near automated methods based on LIDAR (Brenner 2000, Ahlberg 2003, Haala 1999). Some methods require user guidance, for example, to specify the class of geometric objects to fit to the LIDAR data (You 2003, Rou 2002). Completely automatic algorithms tend to generate models with several artifacts (Rottensteiner 2002) due to the presence of noise and resolution limitations of the LIDAR data. Some researchers have approached the problem as a building footprint extraction problem (Alharthy 2000) but their work is limited to fitting flat roof models to LIDAR. In contrast to the previous work on building modeling, the algorithms presented here can be used to automatically extract prismatic building models from LIDAR data.

AUTOMATIC BUILDING EXTRACTION

It is important that the building extraction process be completely automatic for it to scale to large urban area modeling. It is not always possible to find either 2D CAD drawings representing the building footprints or the cadastral maps for all the roofs (Ulm 2003). If cadastral maps are available then it is trivial to estimate the building geometry using LIDAR data. In this work we make no assumption about the building types. In fact, we automatically infer the roof types directly from LIDAR. We model the class of buildings that can be created using a union of simpler parametric roof types shown in Figure 1.

Problem Statement

Given LIDAR data for an urban area, our objective is to automatically extract buildings as a combination of simpler, parametric shapes that can be represented with significantly less amount of storage than the original LIDAR, as well as edited with relative ease in cases where the automatic algorithm is not completely successful. In particular, we would like to achieve the following two goals:

1. Automatically segment roofs and terrain from aerial LIDAR.
2. Infer the shape of the building as a combination of simpler parametric shapes that can be edited easily.

The LIDAR data itself poses several challenges. It can be noisy because of errors in GPS, registration errors, sensor errors, etc. It should also be noted that the LIDAR sensor "sees" only the roofs of the buildings. Hence, information about the geometry of the sides of the buildings is not available. Since we are estimating building shapes using only aerial LIDAR, it is not possible to determine the shape of the sides of the buildings. One can also acquire LIDAR data using a ground-based vehicle to accurately capture the sides of the buildings. Such an approach is taken by Früh (Früh el. al. 2003, 2004). However, their method is limited because they fit a triangle mesh to the LIDAR points and do not simplify the geometry into parametric models. Hence their 3D models are represented by a soup of polygons with more geometry than needed. Such 3D models are not efficient to render. In addition, these 3D models can be quite noisy because the edges of buildings are not straight, walls are not composed of a single plane, etc. These models also lack watertightness.

The benefit of representing buildings as a combination of parametric shapes is that each building can be described by only a few parameters. For example a rectangular box shape building can be described by three parameters (length, width, height) and a gable room building can be described by at most 5 parameters (length, width, height, and two slopes). Another advantage in representing buildings as parametric shapes rather than a soup of polygons is that they can be edited intuitively by operations such as "push a wall", "change the height", "change the slope of gable roof", etc.

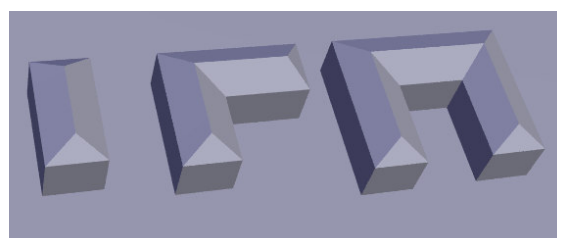

Figure 1. Parametric shapes used to model complex roof types

We propose a novel algorithm for automatic detection of buildings from LIDAR data as a combination of simpler parametric prismatic shapes. The parametric prismatic shapes corresponding to sloped roofs are shown in Figure 1. In addition to sloped roofs we also fit rectangular box shaped buildings. A building that cannot be modeled as one of the parametric shapes shown in Figure 1 is modeled either as a flat or single-sloped roof building with a polygonal base. Our algorithms processes the LIDAR data in the following five stages:

1. Segmentation of points belonging to the roof of a building
2. Approximate roof shape modeling

3. Roof topology inference
4. Identification of parametric sub-parts from the roof topology
5. Parametric model fitting

The following sections discuss these steps in greater detail.

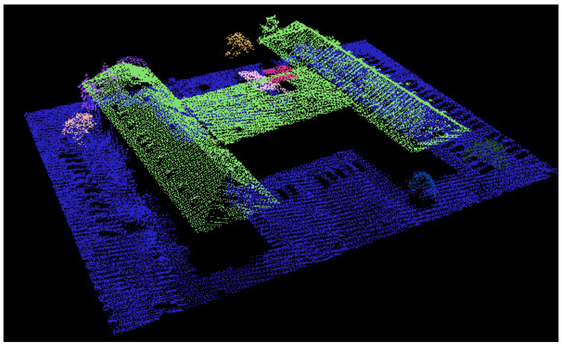

Figure 2. Results of connected components analysis on a LIDAR point cloud. Points are colored according to the connected components to which they belong.

Segmentation

The LIDAR point cloud has points that not only belong to the roofs of buildings but also other surfaces such as terrain, trees, cars, etc. In order to estimate the geometry of the buildings we first need to identify the points that belong to the roofs. Note that the LIDAR data is not available for the sides of the buildings because the LIDAR instrument sees only the roofs of the buildings. Also, the LIDAR point cloud density is expected to be fairly uniform over a given region. Hence, all the points belonging to the roof of a building lie in a single connected component. Since the ground is expected to contain the largest number of points, the largest connected component would contain all the points that belong to the ground. Points belonging to trees and cars would also lie in the same connected component. The LIDAR points do not lie at the corners of a fixed grid, hence we voxelize the volume containing LIDAR points and use a 26-neighbor connected components method to identify points in different connected components. Figure 2 shows the results of connected component analysis on a LIDAR point cloud. Note that points are colored according to the connected component to which they belong. Also note that points belonging to some trees form a separate connected component. These points can be eliminated by using an appropriate threshold because the number of points belonging to the roof are significantly greater then the number of points belonging to the trees.

Approximate roof shape modeling

After all the points belonging to the roof are identified, we build a rough model of the roofs by identifying the individual roof faces. This is achieved by fitting local planar patches to the identified roof points and grouping them together based on the normals of these surface patches. A single plane is fit through each group of points sand their boundary identified using the ball-pivoting algorithm. Figure 3 shows the sequence of steps required to create an approximate roof model.

Roof Topology Inference

Since albeit is desirable to edit the building geometry as parametric shapes, we need to identify the different parametric shapes that can be combined to form the building. Given any roof composed of planar faces, we can describe the relationship between the various faces using a *topological graph* data structure as described below.

- Each face of the roof is represented by a vertex
- Two vertices of the graph have an edge between them if the corresponding polygons are edge-adjacent (they share an edge)

The edges of the graph are labeled as follows depending on the normals of the polygons:

- **O^+:** the polygons corresponding to the end vertices have normals that when projected to the XY plane are orthogonal and point away from each other.
- **O^-**: the polygons corresponding to the end vertices have normals that when projected to the XY plane are orthogonal and point towards each other.
- **S^+**: the polygons corresponding to the end vertices have normals that when projected to the XY plane are parallel and point away from each other.
- **S^-**: the polygons corresponding to the end vertices have normals that when projected to the XY plane are parallel and point towards each other.
- **N**: no constraint

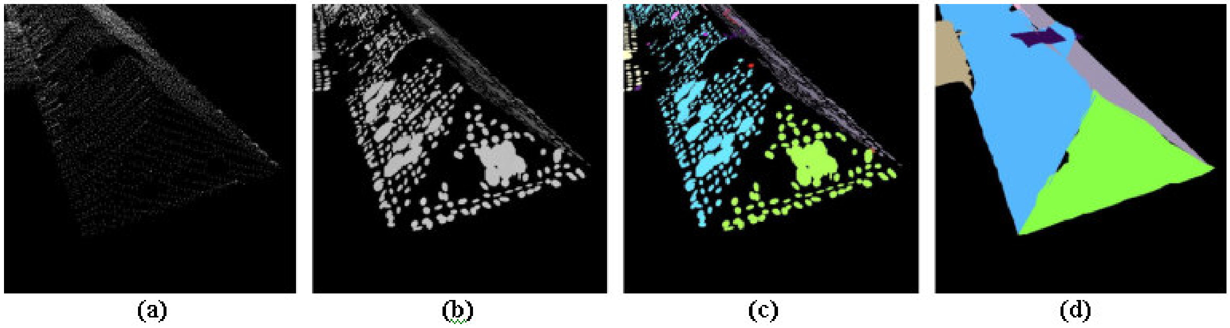

Figure 3. Approximate roof modeling. (a) LIDAR points belonging to the roof. (b) local planar patches fit to the LIDAR points. (c) planar patches grouped together based on similarity of their normals. (d) approximate boundaries of the roof faces.

Figure 4 shows a simple roof (hip roof), and the corresponding topological graph that describes the relationships between the four faces of the roof.

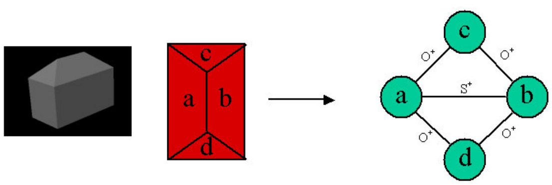

Figure 4. A simple roof and the corresponding topology graph

More complex roofs can similarly be analyzed to determine their topology graphs. Figure 5 shows a complex roof and the corresponding topology graph. The topology graph of a roof is determined by labeling all the faces of the approximate roof as described in the previous section and determining whether two faces share an edge. For each pair of adjacent roof faces we also analyze whether they are orthogonal or symmetric and label the corresponding edge of the topology graph using the labels described above.

Identification of Parametric Sub-parts from the roof topology

It should be noted that the complex roof shown in Figure 5 can be decomposed into multiple simpler roofs as shown in Figure 6-(right). It is also important to note that these simpler roofs can be detected by identifying their topology graphs as sub-graphs of the larger topology graph of the complex roof, i.e. by sub-graph matching. While sub-graph matching is a NP-complete problem, since we expect the graphs to be fairly small, an exhaustive search to do sub-graph matching is quite fast and sufficient for our purpose. The topology graphs corresponding to the roof types in Figure 1 are shown in Figure 7.

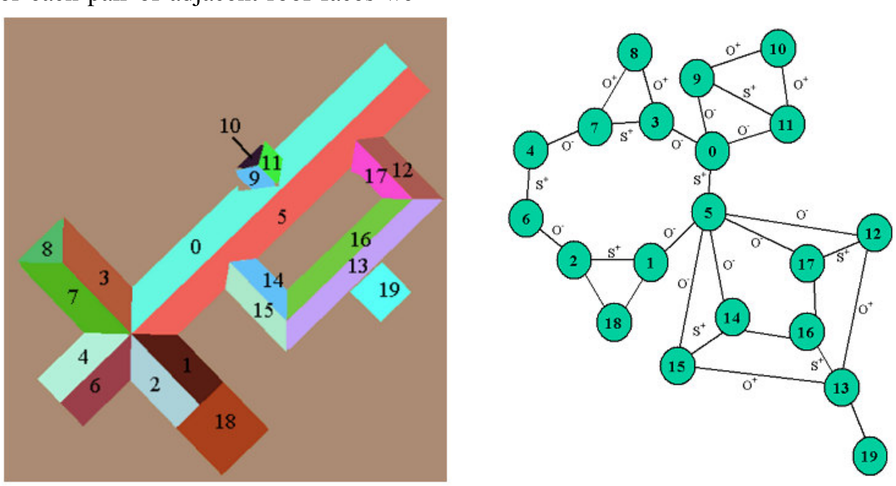

Figure 5. A complex roof and the corresponding topology graph

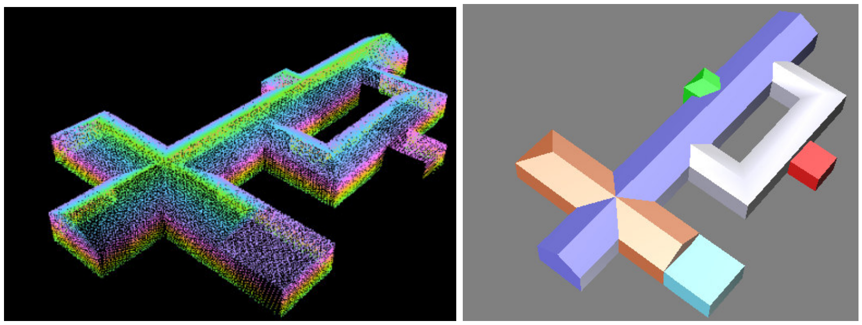

Figure 6. Left: A synthetic LIDAR point cloud. Right: Result of automatic building geometry extraction as a union of the simple roofs shown in Figure 1. Each simple roof is shown with a different color.

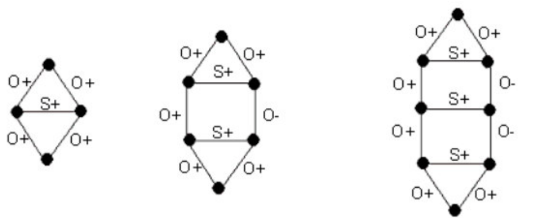

Figure 7. Topology graphs that describe the three roof shapes shown in Figure 1.

Parametric model fitting

Once the various sub-parts of the roof have been identified as one of the parametric shapes, we can directly fit planar patches to the original LIDAR points such that the orthogonality and symmetry constraints specified by the topology graph are observed. Result of this step is shown in Figure 6. Figure 8 shows the result of our automatic building extraction algorithm to a LIDAR point cloud. The LIDAR point cloud and the extracted parametric building models are overlaid to illustrate how well the building geometry fits the LIDAR data.

CONCLUSIONS AND FUTURE WORK

We have presented a novel algorithm to automatically estimate the geometry of buildings from a LIDAR point cloud of an outdoor scene as a combination of simple parametric shapes. The algorithms presented here can be used to automatically model large areas of LIDAR data. The advantage of our method is that it requires no user intervention, would scale well to large areas, and the resulting geometry is in the form of a collection of parametric shapes that can be represented using minimum amount of storage as well as intuitively modified. While we can automatically model complex roofs as a combination of simpler shapes, our system can only handle planar roofs. In the future we would like to extend our algorithms to automatically fit more general surfaces to LIDAR data.

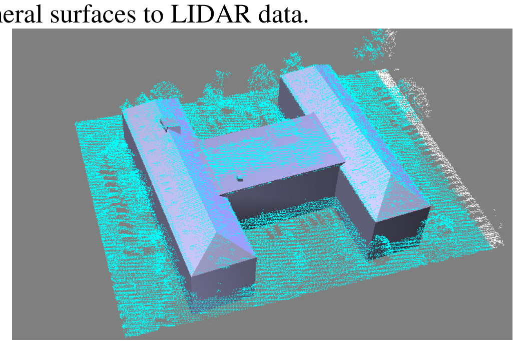

Figure 8. Result of automatic building extraction from LIDAR point cloud. The extracted building and the original point cloud are shown.

ACKNOWLEDGEMENTS

We would also like to thank ONR and PMTRASYS for supporting this research effort (contract number NMA201-01-C-0003).

REFERENCES

Alharthy, A. & Bethel, J. (2002). Heuristic Filtering and 3D Feature Extraction from LIDAR Data. *Photogrammetric Computer Vision Symposium*, 2002. September 9 - 13, 2002, Graz, Austria.

Ahlberg, S., Soderman, U., Elmqvist, M. & Persson, A.. (2004). On Modeling and Visualization of High Resolution Virtual Environments Using LIDAR Data. *Geoinformatica 2004. Proceedings of the 12th International Conference on Geoinformatics*, Sweden, 2004. pp 299 - 306.

Bailey, M. & Armstrong, M. (2004). Science & Technology Long Poles in the Tent. *Marine Corps Perspective on S&T Shortcomings Relative to Technology-Supported Training*. July 2004.

Bauer, J., Karner, K., Schindler, K, Klaus, A. & Zach, C. (2003). Segmentation of Building Models from Dense 3D Point Clouds. *Workshop of the Austrian Association for Pattern Recognition*, Laxenburg, Austria, 2003.

Brenner, C. (2000). Towards Fully Automatic Generation of City Models. ISPRS, vol. XXXIII, Amsterdam, 2000.

Elaksher, A. F. & Bethel, J. S. (2002). Reconstructing 3D Buildings from LIDAR Data. *Photogrammetric Computer Vision Symposium*, September 9 - 13, 2002, Graz, Austria.

Früh, C. & Zakhor, A (2003). Constructing 3D City Models by Merging Ground-Based and Airborne Views. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2003, pp. 562 – 569.

Früh, C. & Zakhor, A. (2004). An Automated Method for Large-Scale, Ground-Based City Model Acquisition. *International Journal of Computer Vision*, Vol. 60, No. 1, October 2004, pp. 5 - 24.

Haala, N. & Brenner, C. (1999). Extraction of Buildings and Trees in Urban Environments. *ISPRS Journal of Photogrammetry & Remote Sensing*, 54 (1999), pp 130 - 137.

Lee, S. C., Huertas, A. & Nevatia, R. (2002). Modeling 3D Complex Buildings with User Assistance. *WACV* 2000, Palm Springs, pp. 170-177, 2002.

Rau, J-Y. & Chen, L-C. (2002). An Interactive System for Producing Polyhedral and Prismatic Building Models Using Building Outline Segments. *Symposium on Geospatial Theory, Processing and Applications*. Ottawa, July 2002

REALViz ImageModeler. www.realviz.com.

Rottensteiner, F. & Jansa, J. (2002). Automatic Extraction of Buildings from LIDAR Data and Aerial Images. *International Archives of Photogrammetry and Remote Sensing*., Ottawa, July 2002.

Ulm, K (2003). Improved 3D city modeling with Cybercity-Modeler (CC-Modeler™). *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *Vol. XXXIV-5/W01,* February 2003.

You, S. Neumann, U. & Fox, P. (2002). "Urban Site Modeling From LIDAR". *Proceedings of Second International Workshop on Computer Graphics and Geometric Modeling*, Vol. 2668, pp. 579 - 588, Montreal, CANADA, May 2003.