3–D Model Construction Using Range and Image Data *

Ioannis Stamos and Peter K. Allen, Columbia University{istamos, allen}@cs.columbia.edu

Abstract

This paper deals with the automated creation of geometric and photometric correct 3-D models of the world. Those models can be used for virtual reality, tele–presence, digital cinematography and urban planning applications. The combination of range (dense depth estimates) and image sensing (color information) provides data–sets which allow us to create geometrically correct, photorealistic models of high quality. The 3-D models are first built from range data using a volumetric set intersection method previously developed by us. Photometry can be mapped onto these models by registering features from both the 3–D and 2–D data sets. Range data segmentation algorithms have been developed to identify planar regions, determine linear features from planar intersections that can serve as features for registration with 2-D imagery lines, and reduce the overall complexity of the models. Results are shown for building models of large buildings on our campus using real data acquired from multiple sensors.

1 Introduction

The recovery and representation of 3–D geometric and photometric information of the real world is one of the most challenging problems in computer vision research. With this work we would like to address the need for highly realistic geometric models of the world, in particular to create models which represent outdoor urban scenes. Those models may be used in applications such as virtual reality, tele-presence, digital cinematography and urban planning.

Our goal is to create an accurate geometric and photometric representation of the scene by means of integrating range and image measurements. The geometry of a scene is captured using range sensing technology whereas the photometry is captured by means of cameras. We have developed a system which, given a set of unregistered depth maps and unregistered photographs, produces a geometric and photometric correct 3–D model representing the scene.

We are dealing with all phases of geometrically correct, photorealistic 3–D model construction with a minimum of human intervention. This includes data acquisition, segmentation, volumetric modeling, viewpoint registration, feature extraction and matching, and merging of range and image data into complete models. Our final result is not just a set of discrete colored voxels or dense range points but a true geometric CAD model with associated image textures.

The entire modeling process can be briefly summarized as follows: 1) Multiple range scans and 2-D images of the scene are acquired. 2) Each range scan is segmented into planar regions (section 3.1). 3) 3-D linear features from each range scan are automatically found (section 3.2). 4) The segmented range data from each scan is registered with the other scans (section 3.3). 5) Each segmented and registered scan is swept into a solid volume, and each volume is intersected to form a complete, 3-D CAD model of the scene (section 4). 6) Linear features are found in each 2-D image using edge detection methods (section 3.4). 7) The 3-D linear features from step 3 and the 2-D linear features from step 6 are matched and used to create a fully textured, geometrically correct 3-D model of the scene (sections 3.5 and 4).

Figure 1 describes the data flow of our approach. We start with multiple, unregistered range scans and photographs of a scene, with range and imagery acquired from different viewpoints. In this paper, the locations of the scans are chosen by the user, but we have also developed an automatic method described in [17] that can plan the appropriate Next Best View. The range data is then segmented into planar regions. The planar segmentation serves a number of purposes. First, it simplifies the acquired data to enable fast and efficient volumetric set operations (union and intersection) for building the 3-D models. Second, it provides a convenient way of identifying prominent 3-D linear features which can be used for registration with the 2-D images. 3–D linear segments are extracted at the locations where the planar faces intersect, and 2–D edges are extracted from the 2–D imagery. Those linear segments (2–D and 3–D) are the features used for the registration between depth maps and between depth maps and 2–D imagery. Each segmented and registered depth map is then transformed into a partial 3–D solid model of the scene using a volumetric sweep method previously developed by us. The next step is

*Supported in part by an ONR/DARPA MURI award N00014-95-1-0601 and NSF grants CDA-96-25374 and EIA-97-29844. Thanks to Michael Reed for his help in model generation.

1063-6919/00 $10.00 © 2000 IEEE

to merge those registered 3–D models into one composite 3–D solid model of the scene. That composite model is then enhanced with 2–D imagery which is registered with the 3–D model by means of 2–D and 3–D feature matching.

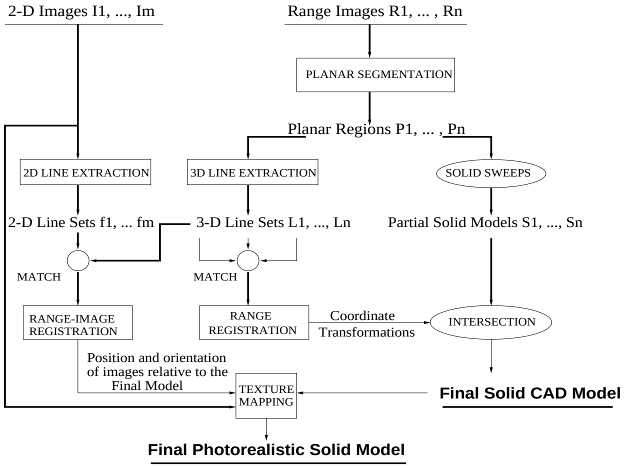
2-D Images I1, ..., Im
Range Images R1, ... , Rn
PLANAR SEGMENTATION
Planar Regions P1, ... , Pn
2D LINE EXTRACTION
3D LINE EXTRACTION
SOLID SWEEPS
2-D Line Sets f1, ... fm
3-D Line Sets L1, ..., Ln
Partial Solid Models S1, ..., Sn
MATCH
MATCH
RANGE-IMAGE REGISTRATION
RANGE REGISTRATION
Coordinate Transformations
INTERSECTION
Position and orientation of images relative to the Final Model
TEXTURE MAPPING
Final Solid CAD Model
Final Photorealistic Solid Model

Figure 1: System for building geometric and photometric correct solid models.

2 Related work

The extraction of photorealistic models of outdoor environments has received much attention recently including an international workshop [11]. Notable work includes the work of Shum et al. [19], Becker [1] and Debevec et al. [6]. These methods use only 2–D images and require a user to guide the 3-D model creation phase. This leads to lack of scalability wrt the number of processed images of the scene or to the computation of simplified geometric descriptions of the scene which may not be geometrically correct. Teller [21, 4] uses an approach that acquires and processes a large amount of pose–annotated spherical imagery of the scene. This imagery is registered and then stereo methods are used to recreate the geometry. Zisserman's group in Oxford [8] works towards the fully automatic construction of graphical models of scenes when the input is a sequence of closely spaced 2–D images (video sequence). Both of the previous methods provide depth estimates which depend on the texture and geometric structure of the scene and which may be quite sparse.

Our approach differs in that we use range sensing to provide dense geometric detail which can then be registered and fused with images to provide photometric detail. It is our belief that using 2-D imagery alone (i.e. stereo methods) will only provide sparse and unreliable geometric measures unless some underlying simple geometry is assumed. A related project using both range and imagery is the work of the VIT group [23, 2].

The broad scope of this problem requires us to use range image segmentation [3, 10], 3–D edge detection [12, 15], 3–D Model Building [5, 22] and image registration methods [13] as well.

3 System Description

In our previous research, we have developed a method which takes a small number of range images and builds a very accurate 3-D CAD model of an object (see [18, 16] for details). The method is an incremental one that interleaves a sensing operation that acquires and merges information into the model with a planning phase to determine the next sensor position or "view". The model acquisition system provides facilities for range image acquisition, solid model construction and model merging: both mesh surface and solid representations are used to build a model of the range data from each view, which is then merged with the model built from previous sensing operations.

For each range image a solid CAD model is constructed. This model consists of sensed and extruded (along the sensing direction) surfaces. The sensed surface is the boundary between the inside and the outside (towards the sensor) of the sensed object. The extruded surfaces are the boundary between empty sensed 3-D space and un-sensed 3-D space. The merging of registered depth maps is done by means of volumetric boolean set intersection between their corresponding partial solid models

We have extended this method to building models of large outdoor structures such as buildings. We are using a Cyrax range scanner which has centimeter level accuracy at distances up to 100 meters to provide dense range sampling of the scene. However, these point data sets can be extremely large (1K x 1K) and computationally unwieldy. The buildings being modeled, although geometrically complex, are comprised of many planar regions, and for reasons of computational and modeling efficiency, these dense point sets can be abstracted into sets of planar regions. In section 4 we present our results of extending this method of automatically building 3-D models to buildings with significant geometry. First we discuss our segmentation methods.

3.1 Planar Segmentation of Range Data

We group the 3–D points from the range scans into clusters of neighboring points which correspond to the same planar surface. In the *Point Classification* phase a plane is fit to the points $\mathbf{v_i}$ which lie on the $k \times k$ neighborhood of every point P. If the deviation of the points from the fitted plane is below a user specified threshold P_{thresh} the center of the neighborhood is classified as *locally planar* point.

A list of clusters is initialized, one cluster per *locally planar* point. The next step is to merge the initial list of clusters and to create a minimum number of clusters of maximum size. Each cluster is defined as a set of 3–D points which are connected and which lie on the same algebraic surface (plane in our case). We visit all locally-planar points P and its neighbors A_j in a manner identical to the sequential labeling algorithm. Two adjacent *locally planar* points are considered to lie on the same planar surface if their corresponding local planar patches have similar orientation and are close in 3D space.

We introduce a metric of co–normality and co–planarity of two planar patches which have been fit around locally planar points P_1 and P_2. The normal of the patches are $\mathbf{n_1}$ and $\mathbf{n_2}$ respectively. The two planar patches are considered to be part of the same planar surface if a) they have identical orientation (within a tolerance region), that is the angle $\alpha = \cos^{-1}(\mathbf{n_1} \cdot \mathbf{n_2})$ is smaller than a threshold α_{thresh} [co–normality measure] and b) they lie on the same infinite plane [co–planarity measure]. The distance between the two patches $d = \max(|\mathbf{r_{12}} \cdot \mathbf{n_1}|, |\mathbf{r_{12}} \cdot \mathbf{n_2}|)$ should be smaller than a threshold d_{thresh} ($\mathbf{r_{12}}$ is the vector connecting the projections of P_1 and P_2 onto their corresponding local planes).

Finally we fit a plane on all points of the final clusters. We also extract the *outer boundary* of this plane, *the convex hull* of this boundary and the axis-aligned three-dimensional *bounding box* which encloses this boundary. These are used for fast distance computation between the extracted bounded planar regions which is described in the next section.

Figure 2a shows a point data set from a range scan of a building on our campus by a Cyrax scanner from a single view (992 by 989 points). If this data is triangulated into facets, it creates an extremely large, unwieldy, and unnecessarily complex description of an object that is composed of many tiny, contiguous planar surfaces. Our algorithm tries to recover and preserve this structure while effectively reducing the data set size and increasing computational efficiency. Figure 2b shows the segmentation of this data set into planar regions, resulting in a large data reduction - approximately 80% less triangular facets (reduction from $1,005,004$ to $207,491$ range points) are needed to model this data set. The parameters used were $P_{thresh} = 0.08$, $\alpha_{thresh} = 0.04$ degrees and $d_{thresh} = 0.01$ meters. The size of the neighborhood used to fit the initial planes was 7 by 7.

3.2 3–D Line Detection

The segmentation also provides a method of finding prominent linear features in the range data sets. The intersection of the planar regions provides three dimensional lines which can be used both for registering multiple range images and matching 3-D lines with 2-D lines from imagery. Extraction of 3-D lines is done in three stages.

First, we compute the infinite 3–D lines at the intersection of the extracted planar regions. We do not consider every possible pair of planar regions but only those whose three-dimensional bounding boxes are close wrt each other (distance threshold d_{bound}). The computation of the distance between two bounding boxes is very fast. However this measure may be inaccurate. Thus we may end up with lines which are the intersection of non-neighboring planes.

The next step is to filter out fictitious lines which are produced by the intersection of non-neighboring planes. We disregard all lines whose distance from both producing polygons is larger than a threshold d_{poly}. The distance of the 3–D line from a convex polygon (both the line and the polygon lie on the same plane) is the minimum distance of this line from every edge of the polygon. In order to compute the distance between two line segments we use a fast algorithm described in [14].

Finally we need to keep the parts of the infinite 3–D lines which are verified from the data set (that is we extract linear segments out of the infinite 3–D lines). We compute the distance between every point of the clusters Π_1 & Π_2 and the line L (Π_1 & Π_2 are the two neighboring planar clusters of points whose intersection produces the infinite line L). We then create a list of the points whose distance from the line L is less than d_{poly} (see previous paragraph). Those points (points which are close wrt the limit d_{poly} to the line) are projected on the line. The linear segment which is bounded by those points is the final result.

Figure 2c shows 3-D lines recovered using this method and Figure 2d shows these lines overlaid on the 2-D image of the building after registering the camera's viewpoint with the range data.

3.3 Registering Range Data

To create a complete description of a scene we need to acquire and register multiple range images. The registration (computation of the rotation matrix R and translation vector $\mathbf{T}$) between the coordinate systems of the n_{th} range image (C_n) and the first image (C_0) is done when a matched set of 3–D features of the n_{th} and first image are given. The 3–D features we use are infinite 3–D lines which are extracted using the algorithm described in the previous section. A linear feature f is represented by the pair of vectors $(\mathbf{n}, \mathbf{p})$, where $\mathbf{n}$ is the direction of the line and $\mathbf{p}$ a point on the line. A solution for the rotation and translation is possible when

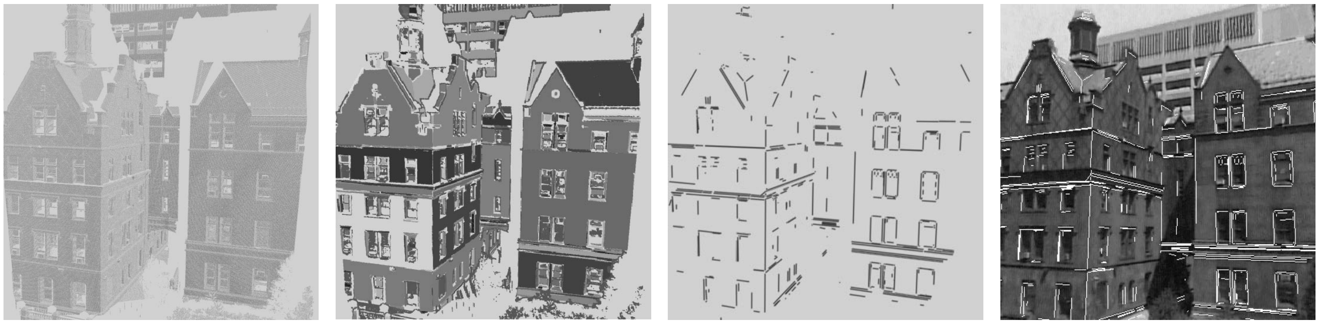

Figure 2: a) Range data scan, b) Planar Segmentation (different planes correspond to different colors), c) Extracted 3D lines generated from the intersection of planar regions, d) 3-D lines projected on the image after registration.

at least two line matches are given. The minimization of the error function $\Sigma||\mathbf{n_i}' - R\mathbf{n_i}||^2$ (where $\mathbf{n_i}', \mathbf{n_i}$ is the i_th pair of matched line orientations between the two coordinate systems) leads to a closed form solution of the rotation (expressed as a quaternion) [7].

Given the solution for the rotation we solve for the translation vector $\mathbf{T}$. We establish a correspondence between two arbitrary points on line $< \mathbf{n}, \mathbf{p} >$, $\mathbf{p_j} = \mathbf{p} + t_j\mathbf{n}$ and two points on its matched line $< \mathbf{n}', \mathbf{p}' >$, $\mathbf{p_j}' = \mathbf{p}' + t_j'\mathbf{n}'$. Thus we have two vector equations $\mathbf{p_j}' = R\mathbf{p_j} + \mathbf{T}$ which are linear in the three components of the translation vector and the four unknown parameters (t_j, t_j') (2x3 linear equations and 7 unknowns). At least two matched lines provide enough constraints to solve the problem through the solution of a linear over-constrained system of equations. Results of the registration are presented in section 4.

3.4 2–D Line Detection

In order to compute 2–D linear image segments we apply the Canny edge detection algorithm with hysteresis thresholding. That provides chains of 2–D edges where each edge is one pixel in size (edge tracking). We used the program *xcv* of the TargetJr distribution [20] in order to compute the Canny edges. The next step is the segmentation of each chain of 2–D edges into linear parts. Each linear part has a minimum length of l_{min} edges and the maximum least square deviation from the underlying edges is n_{thresh}. The fitting is incremental, that is we try to fit the maximum number of edges to a linear segment while we traverse the edge chain (orthogonal regression).

3.5 Registering Range with Image Data

We now describe the fusion of the information provided by the range and image sensors. These two sensors provide information of a qualitatively different nature and have distinct projection models. The fusion of information between those two sensors requires the knowledge of the internal camera parameters (effective focal length, principal point and distortion parameters) and the relative position and orientation between the centers of projection of the camera and the range sensor. The knowledge of those parameters allows us to invert the image formation process and to project back the color information captured by the camera on the 3–D points provided by the range sensor. Thus we can create a photorealistic representation of the environment.

The estimation of the unknown position and orientation of an internally calibrated camera wrt the range sensor is possible if a corresponding set of 3–D and 2–D features is known. Currently, this corresponding set of feature matches is provided by the user but our goal is its automatic computation (see section 5 for a proposed method). We adapted the algorithm proposed by Kumar & Hanson [13] for the registration between range and 2–D images, when a set of corresponding 3–D and 2–D line pairs is given and the internal parameters of the camera are known.

Let $\mathbf{N_i}$ be the normal of the plane formed by the ith image line and the center of projection of the camera. This vector is expressed in the coordinate system of the camera. The sum of the squared perpendicular distance of the endpoints $\mathbf{e_i^1}$ and $\mathbf{e_i^2}$ of the corresponding ith 3–D line from that plane is $d_i = (\mathbf{N_i} \cdot (R(\mathbf{e_i^1}) + \mathbf{T}))^2 + (\mathbf{N_i} \cdot (R(\mathbf{e_i^2}) + \mathbf{T}))^2$, where the endpoints $\mathbf{e_i^1}$ and $\mathbf{e_i^2}$ are expressed in the coordinate system of the range sensor. The error function we wish to minimize wrt the rotation matrix R and the translation vector $\mathbf{T}$ is $E_1(R, \mathbf{T}) = \Sigma_{i=1}^N d_i$. This error function expresses the perpendicular distance of the endpoints of a 3–D line from the plane formed by the perspective projection of the corresponding 2–D line into 3–D space. In the next section we will present results for the complete process of integrating multiple views of both range and imagery.

4 Results

In this section we present results of the complete process: 3–D model acquisition, planar segmentation, volumetric sweeping, volume set intersection, range registration, and range and image registration for a large, geometrically complex and architecturally detailed building on our campus. The building is pictured in Figure 3i.

Three range scans of the building were acquired at a resolution of 976 by 933 3D points. Figures 3a-c show each of these scans, with each data point colored according to the planar region it belongs to from the segmentation process (color images at `http://www.cs.columbia.edu/robotics`). The parameters used for the segmentation were $P_{thresh} = 0.08$, $\alpha_{thresh} = 0.04$ degrees and $d_{thresh} = 0.08$ meters. Figure 3d shows the integrated range data set after registering the range images. The range images were registered by manually selecting corresponding 3-D lines generated by the method in section 3.3.

Figures 3e-g show the volumetric sweeps of each segmented range scan. These sweeps are generated by extruding each planar face of the segmented range data along the range sensing direction. Volumetric set intersection is then used to intersect the sweeps to create the composite geometric model which encompasses the entire building and is shown in Figure 3h.

Figure 3j shows one of the 2-D images of the building (the image shown in Figure 3i) texture-mapped on the composite model of Figure 3h. We are calculating the relative position and orientation of the camera wrt to the model as follows: a) the user selects a set of automatically extracted 2–D lines on the 2–D image and its corresponding set of 3–D lines which have also been automatically extracted from the individual depth maps and b) the algorithm described in section 3.5 provides the extrinsic parameters of the camera wrt the 3–D model. The internal camera parameters were approximated in this example.

5 Discussion

We have described a system for building geometrically correct, photorealistic models of large outdoor structures. The system is comprehensive in that includes modules for data acquisition, segmentation, volumetric modeling, feature detection, registration, and fusion of range and image data to create the final models. The final models are true solid CAD models with texture mapping and not just depth maps or sets of colored voxels. This allows these models to be more easily used in many upstream applications, and also allows modeling a wide range of geometry.

We would like to extend the system towards the direction of minimal human interaction. At this point the human is involved in two stages: a) the internal calibration of the camera sensor and b) the selection of the matching set of 3–D and 2–D features. We have implemented a camera self–calibration algorithm when three directions of parallel 3–D lines are detected on the 2–D image based on [1]. The automated extraction of lines of this kind is possible in environments of man–made objects (e.g. buildings). More challenging is the automated matching between sets of 3–D and 2–D features. Again the extraction of three directions of parallel 3–D lines (using the automated extracted 3–D line set) and the corresponding directions of 2–D lines (using the automated extracted 2–D line set) can be the first step in that procedure.

As a final step in automating this process, we have also built a mobile robot system which contains both range and image sensors which can be navigated to acquisition sites to create models (described in [9]).

References

[1] S. C. Becker. *Vision–assisted modeling from model–based video representations.* PhD thesis, Massachusetts Institute of Technology, Feb. 1997.

[2] J.-A. Beraldin, L. Cournoyer, et al. Object model creation from multiple range images: Acquisition, calibration, model building and verification. In *Intern. Conf. on Recent Advances in 3–D Dig. Imaging and Modeling*, pages 326–333, Ottawa, Canada, May 1997.

[3] P. J. Besl and R. C. Jain. Segmentation through variable–order surface fitting. *IEEE Trans. on PAMI*, 10(2):167–192, Mar. 1988.

[4] S. Coorg and S. Teller. Extracting textured vertical facades from contolled close-range imagery. In *CVPR*, pages 625–632, Fort Collins, Colorado, 1999.

[5] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *SIGGRAPH*, pages 303–312, 1996.

[6] P. E. Debevec, C. J. Taylor, and J. Malik. Modeling and rendering architecure from photographs: A hybrid geometry-based and image-based approach. In *SIGGRAPH*, 1996.

[7] O. Faugeras. *Three–Dimensional Computer Vision.* The MIT Press, 1996.

[8] A. W. Fitzgibbon and A. Zisserman. Automatic 3D model acquisition and generation of new images from video sequences. In *Proc. of European Signal Processing Conf. (EUSIPCO '98), Rhodes, Greece*, pages 1261–1269, 1998.

[9] A. Gueorguiev, P. K. Allen, E. Gold, and P. Blair. Design, architecture and control of a mobile site modeling robot. In *Intern. Conf. on Rob. & Aut.*, San Fransisco, Apr. 2000.

[10] A. Hoover, G. Jean-Baptise, X. Jiang, et al. An experimental comparison of range image segmentation algorithms. In *IEEE Trans. on PAMI*, pages 1–17, July 1996.

[11] Institute of Industrial Science(IIS), The Univers. of Tokyo. *Urban Multi–Media/3D Mapping workshop*, Japan, 1999.

[12] X. Jiang and H. Bunke. Edge detection in range images based on scan line approximation. *Computer Vision and Image Understanding*, 73(2):183–199, Feb. 1999.

[13] R. Kumar and A. R. Hanson. Robust methods for estimating pose and a sensitivity analysis. *Computer Vision Graphics and Image Processing*, 60(3):313–342, Nov. 1994.

[14] V. J. Lumelsky. On fast computation of distance between line segments. *Information Processing Letters*, 21:55–61, 1985.

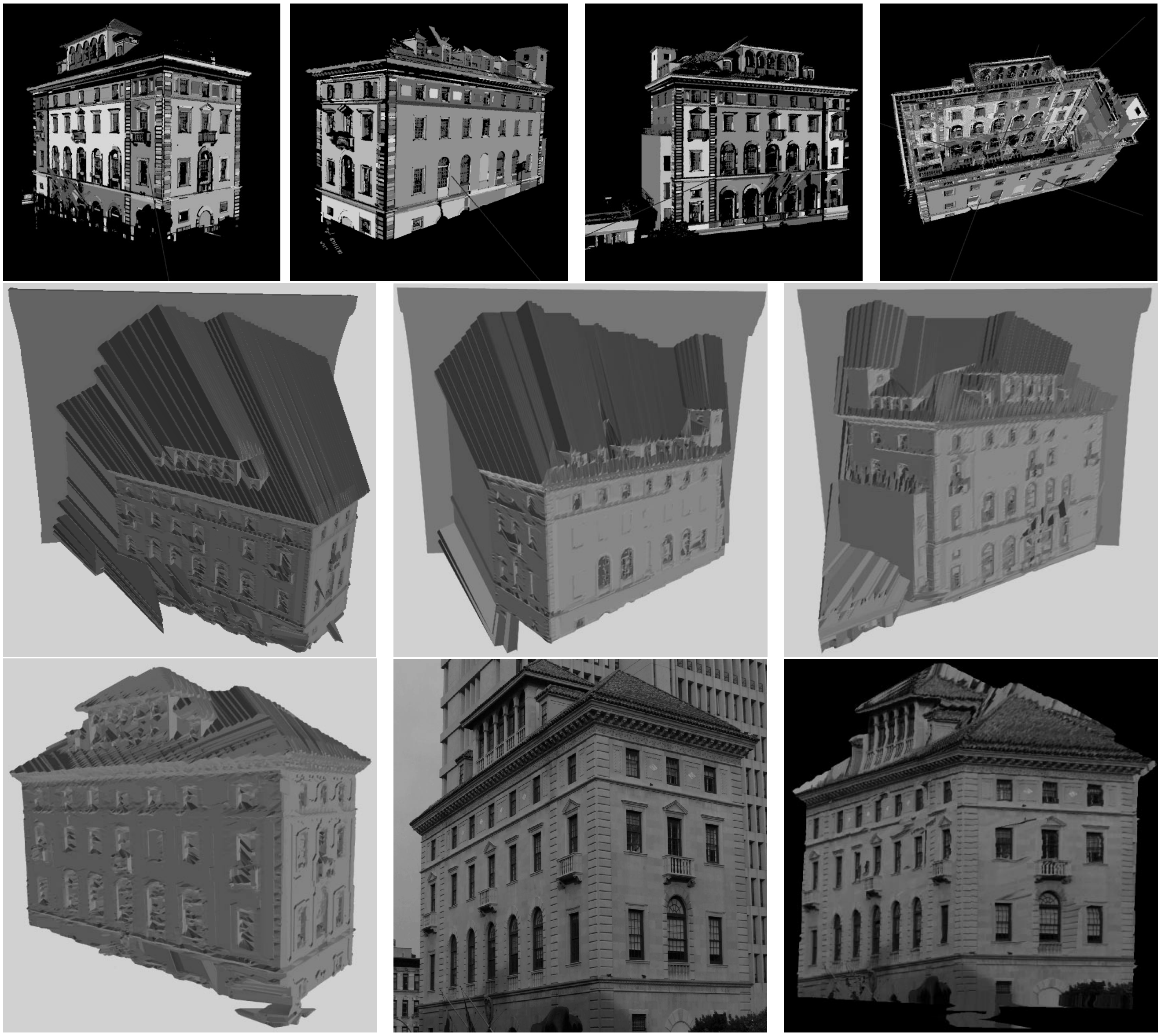

Figure 3: Model Building Process: a) First segmented depth map, b) Second segmented depth map, c) Third segmented depth map, d) Registered depth maps, e) First volumetric sweep, f) Second volumetric sweep, g) Third volumetric sweep, h) Composite solid model generated by the intersection of the three sweeps, i) 2-D image of building, j) Texture-mapped composite solid model.

[15] O. Monga, R. Deriche, and J.-M. Rocchisani. 3D edge detection using recursive filtering: Application to scanner images. *Computer Vision Graphics and Image Processing*, 53(1):76–87, Jan. 1991.

[16] M. Reed and P. K. Allen. 3-D modeling from range imagery. *Image and Vision Computing*, 17(1):99–111, February 1999.

[17] M. Reed and P. K. Allen. Constraint-based sensor planning for scene modeling. In *Computational Intelligence in Robotics and Automation (CIRA99)*, Nov. 1999.

[18] M. Reed, P. K. Allen, and I. Stamos. Automated model acquisition from range images with view planning. In *Computer Vision and Pattern Recognition Conference*, pages 72–77, June 16-20 1997.

[19] H.-Y. Shum, M. Han, and R. Szeliski. Interactive construction of 3D models from panoramic mosaic. In *CVPR*, Santa Barbara, CA, June 1998.

[20] TargetJr. http://www.esat.kuleuven.ac.be/~targetjr/.

[21] S. Teller, S. Coorg, and N. Master. Acquisition of a large pose-mosaic dataset. In *CVPR*, pages 872–878, Santa Barbara, CA, June 1998.

[22] G. Turk and M. Levoy. Zippered polygon meshes from range images. In *SIGGRAPH*, 1994.

[23] Visual Information Technology Group, Canada. http://www.vit.iit.nrc.ca/VIT.html.