Geometry and Texture from Thousands of Images

J.P. Mellor*

Rose-Hulman Institute of Technology, Terre Haute, IN 47803
j.p.mellor@rose-hulman.edu

Abstract. This paper presents a novel method for automatically recovering dense *surface patches* using large sets (1000's) of calibrated images taken from arbitrary positions within the scene. Physical instruments, such as Global Positioning System (GPS), inertial sensors, and inclinometers, are used to estimate the position and orientation of each image. Some of the most important characteristics of our approach are that it: 1) uses and refines noisy calibration estimates; 2) compensates for large variations in illumination; 3) tolerates significant soft occlusion (e.g. tree branches); and 4) associates, at a fundamental level, an estimated normal (eliminating the frontal-planar assumption) and texture with each surface patch.

1 Introduction

The problem of recovering three-dimensional information from a set of photographs or images is essentially the correspondence problem: *Given a point in one image, find the corresponding point in each of the other images*. Typically, photogrammetric approaches (Section 1.1) require manual identification of correspondences, while computer vision approaches (Section 1.2) rely on automatic identification of correspondences. If the images are from nearby positions and similar orientations (short baseline), they often vary only slightly, simplifying the identification of correspondences. Once sufficient correspondences have been identified, solving for the depth is simply a matter of geometry.

1.1 Photogrammetry

A number of the recent interactive modeling systems are based upon photogrammetry. Research projects such as RADIUS [2] and commercial systems such as FotoG are commonly used to extract three-dimensional models from images. Good results have been achieved with these systems, however the requirement for human input limits the size and complexity of the recovered model. One

* This paper describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology and was supported in part by the Advanced Research Projects Agency of the Department of the Defense under Office of Naval Research contract N00014-91-J-4038 and Rome Lab contract F3060-94-C-0204.

approach to reducing the amount of human input is to exploit geometric constraints. The geometric structure typical of urban environments can be used to constrain the modeling process such as Becker and Bove [1], Shum *et al.*[14], and Debevec *et al.* [7]. In spite of this reduction, each image must be processed individually by a human to produce a three-dimensional model, making it difficult to extend these systems to large sets of images. The major strength of these systems is the textured three-dimensional model produced.

1.2 Computer Vision

One approach to automatically solving the correspondence problem is to use multiple images such as Collins [3] and Seitz and Dyer [13] (neither of these approaches is suitable for reconstructions using images acquired from within the scene) or Kutulakos and Seitz [9, 8] (which is not well suited to images acquired in outdoor urban environments).

1.3 Discussion

Photogrammetric methods produce high quality models, but require human input. Computer vision techniques function automatically, but generally do not produce usable models, operate on small sets of images and frequently are fragile with respect to occlusion and changes in illumination [1]. The work presented here draws from both photogrammetry and computer vision. Like photogrammetric methods we produce high quality textured models and like computer vision techniques our method is fully automatic.

Our approach is valid for arbitrary camera positions within the scene and is capable of analyzing very large sets of images. Our focus is recovering built geometry (architectural facades) in an urban environment. However, the algorithms presented are generally applicable to objects that can be modeled by small planar patches. Surface patches (geometry and texture) or *surfels* are recovered directly from the image data. In most cases, three-dimensional position and orientation can be recovered using purely local information, avoiding the computational costs of global constraints. Some of the significant characteristics of this approach are:

- Large sets of images contain both long and short baseline images and exhibit the benefits of both (accuracy and ease of matching). It also makes our method robust to sensor noise and occlusion, and provides the information content required to construct complex models.
- Each image is calibrated - its position and orientation in a single global coordinate system is estimated. The use of a global coordinate system allows data to be easily merged and facilitates geometric constraints. The camera's internal parameters are also known.

[1] For a more complete discussion of related work see [10].

– The fundamental unit is a textured surface patch and matching is done in three-dimensional space. This eliminates the need for the frontal-planar assumption made by many computer vision techniques and provides robustness to soft occlusion (e.g. tree branches). Also, the surface patches are immediately useful as a rough model and readily lend themselves to aggregation to form more refined models. The textured surface patch or *surface element* is referred to as a surfel. This definition differs from Szeliski's [15] in that it refers to a finite sized patch which includes both geometry and texture.
– The algorithm tolerates significant noise in the calibration estimates and produces updates to those estimates.
– The algorithm corrects for changes in illumination. This allows raw image properties (e.g. pixel values) to be used, avoiding the errors and ambiguities associated with higher level constructs such as edges or corners.
– The algorithm scales well. The initial stage is completely local and scales linearly with the number of images. Subsequent stages are global in nature, exploit geometric constraints, and scale quadratically with the complexity of the underlying scene [2].

Not all of these characteristics are unique, but their combination produces a novel method of automatically recovering three-dimensional geometry and texture from large sets of images.

1.4 City Scanning Project

The work presented in this paper is part of the MIT City Scanning project whose primary focus is the Automatic Population of Geospatial Databases (APGD). A self contained image acquisition platform called *Argus* is used to acquire calibrated images [6]. At each location or node images are acquired in a hemispherical tiling. The position and orientation estimates obtained during the acquisition phase are good, but contain a significant amount of error. The estimates are refined using techniques described in [5, 4]. For a more complete description of the project see [16, 17].

1.5 Overview

Figure 1 shows an overview of the reconstruction pipeline described in this paper. The calibrated imagery described above serves as input to the pipeline. The left hand column shows the major steps of our approach; the right hand side shows example output at various stages. The output of the pipeline is a textured three-dimensional model. Our approach can be generally characterized as *hypothesize and test*. We hypothesize a surfel and then test whether it is consistent with the data. Section 2 describes the dataset used for this paper. Section 3 briefly reviews our approach. Section 4 introduces several techniques to remove false

[2] This is the worst case complexity. With spatial hashing the expected complexity is linear in the number of reconstructed surfels.

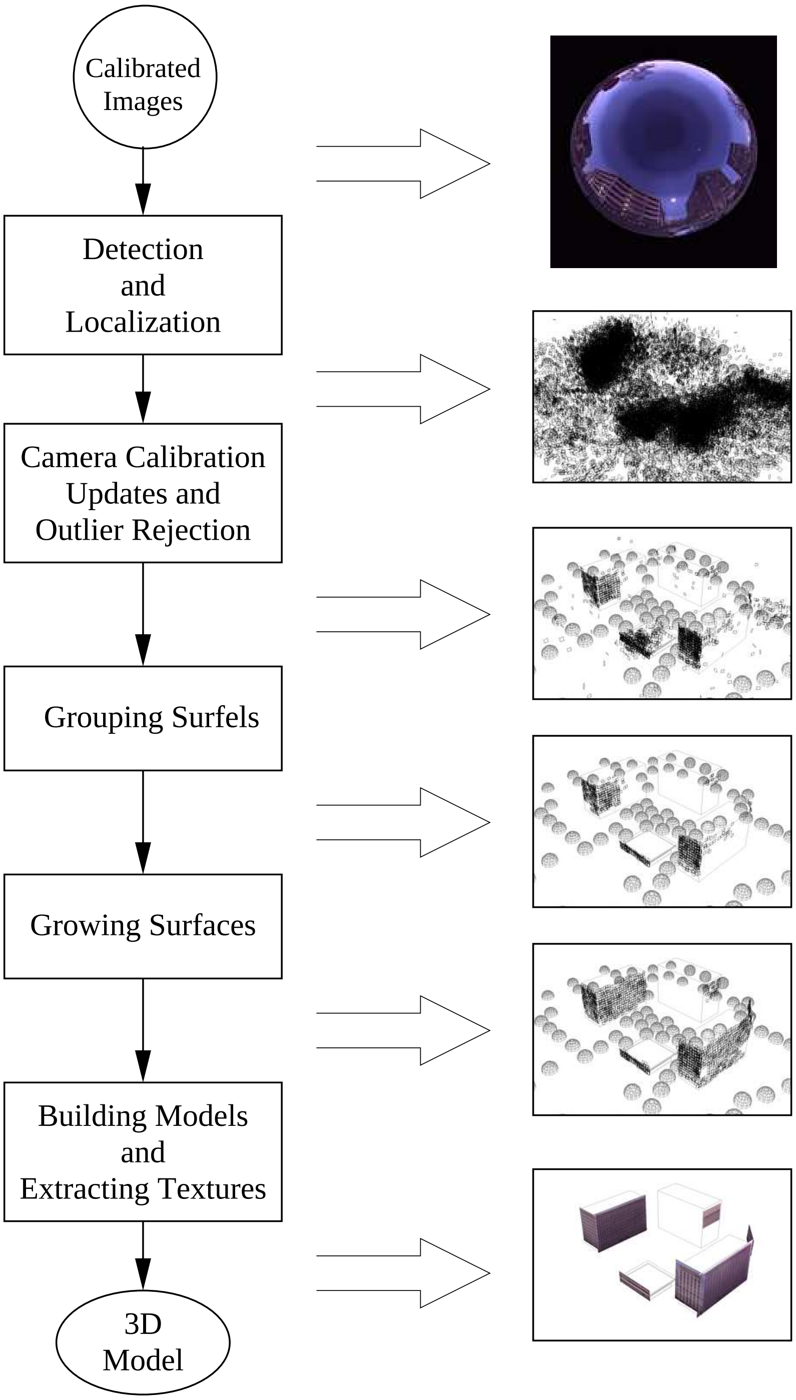


Fig. 1. Overview.

positives and fill in the missing parts. Finally we discuss fitting simple models and extracting textures. We present the results of applying our method to a large dataset.

2 The Dataset

A Kodak DCS 420 digital camera mounted on an instrumented platform was used to acquire a set of calibrated images in and around Technology Square [6]. Nearly 4000 images were collected from 81 node points. Other than avoiding inclement weather and darkness, no restrictions were placed on the day and time of, or weather conditions during, acquisition. The location of each node is shown in figure 2. At each node, the camera was rotated about the focal point collecting images in a hemispherical mosaic. Most nodes are tiled with 47 images. The raw images are 1524×1012 pixels and cover a field of view of $41° \times 28°$. Each node contains approximately 70 million pixels. After acquisition, the images are reduced to quarter resolution (381×253 pixels) and mosaiced [5, 4]. Equal area projections of the spherical mosaic from two nodes are shown in Figure 3. The node on the left was acquired on an overcast day and has a distinct reddish tint. The one on the right was acquired on a bright clear day. Significant shadows are present in the right image whereas the left has fairly uniform lighting. Following mosaicing, the estimated camera calibrations are refined.

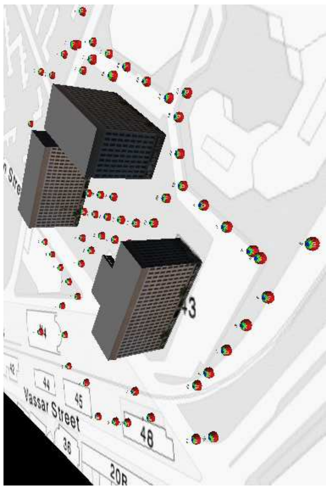


Fig. 2. Node locations.

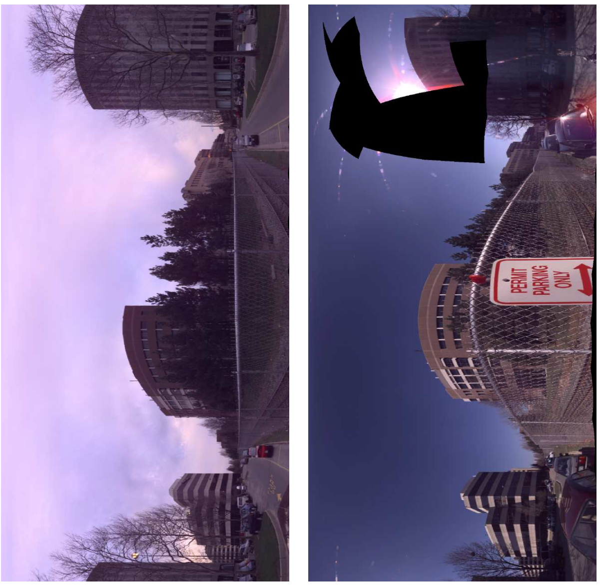


Fig. 3. Example nodes.

After refinement, the calibration data is good, but not perfect. The pose estimates are within about 1° and 1 meter of the actual values. These errors produce an offset between corresponding points in different images. A 1° pose error will displace a feature by over 8 pixels. Our calibration estimates are in an absolute coordinate frame which allows us to integrate images regardless of when or from what source they were collected. This greatly increases the quantity and quality of available data, but because of variations in illumination condition also complicates the analysis.

Fig. 4. Reprojection onto surfel 1 coincident with actual surface.

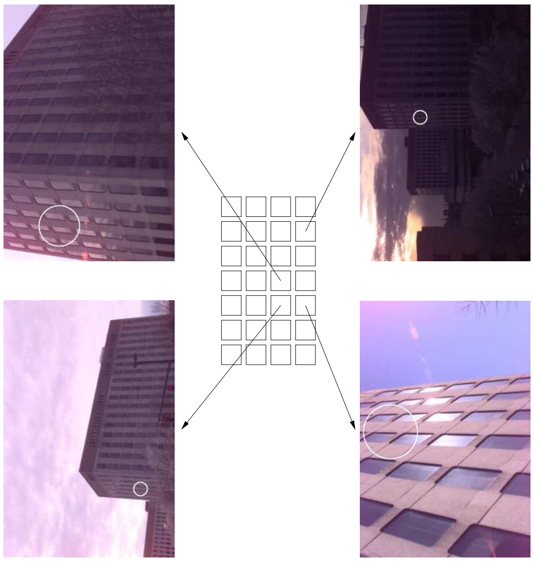


Fig. 5. Source images for selected regions of surfel 1.

Figures 4 and 6 show several images from our dataset reprojected (using the estimated camera calibration) onto a surfel which is coincident with an actual surface. The location, orientation, and size of the surfels used are shown in Table 1. Surfel 1 was used to generate the collection of images in Figure 4 and surfel 2 those in Figure 6. If the camera calibration estimates were perfect and the illumination was constant, the regions in each figure should (ignoring errors introduced during image formation and resampling) be identical. The misalignment present in both sets is the result of error in the calibration estimates. Figure 4 is representative of the best in the dataset. A large number of the source images have high contrast and none of the regions are occluded. The third row has a distinct reddish tint. The four images in the center of the last row were collected under direct sunlight. And, the last two images were taken near dusk.

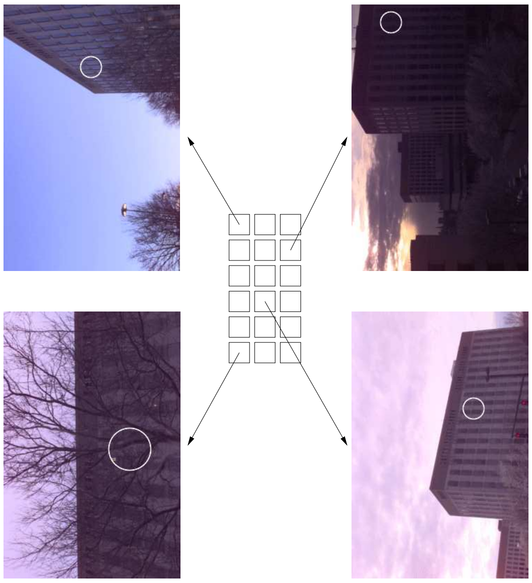


Fig. 6. Reprojection onto surfel 2 coincident with actual surface.

Fig. 7. Source images for selected regions of surfel 2.

Figure 6 is more typical of the dataset. It is lower in contrast and some of the regions are partially occluded by trees. Figures 5 and 7 show source images with the reprojected area marked by a circle for several of the regions shown in Figures 4 and 6. In the worst cases all views of a surfel are similar to the upper left image of Figure 7.

3 Images to Surfels

Ideally, if a hypothesized surfel is coincident with an actual surface, reprojecting images onto the surfel should produce a set of regions which are highly correlated. On the other hand, if the surfel is not coincident with a surface, the subimages should not be correlated. This is the basic idea behind our approach [11]. As shown in Figures 4 and 6, noisy data is not quite this simple and we must extend our algorithm to handle camera calibration error, significant variation in illumination condition, and image noise (e.g. soft occlusion from tree branches) [12]. To compensate for camera calibration error we allow the source images to be shifted in the image plane prior to reprojection. We use optimization techniques to find the best alignment and the maximum shift is a function of the bound on calibration error. Illumination condition is normalized using a linear correction for each color channel. Finally, noisy pixels in the reprojection may individually be rejected as outliers. Figures 8 and 9 show the regions from Figures 4 and 6 after compensating for camera calibration error and illumination condition.

Surfel	Position			Normal			Size (units)		Size (pixels)	
	x	y	z	x	y	z	x	y	x	y
1	-1445	-2600	1200	-1	0	0	110	110	11	11
2	-280	-3078	1500	0	-1	0	110	110	11	11

Table 1. Surfel parameters.

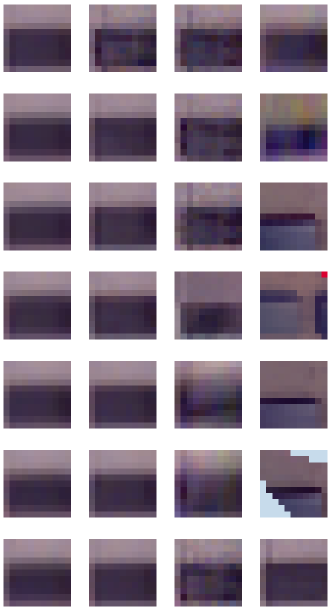


Fig. 8. Aligned and corrected regions for surfel 1.

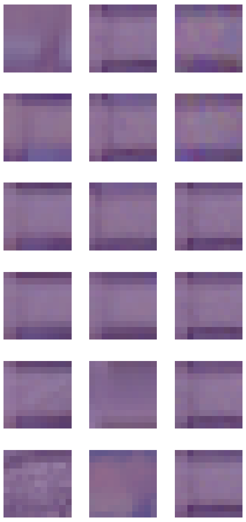


Fig. 9. Aligned and corrected regions for surfel 2.

Our extensions to handle camera calibration error, significant variation in illumination condition, and image noise add additional degrees of freedom to the problem making over-fitting a concern. Section 4 introduces several geometric constraints to prune false positives. One beneficial side effect is that surfels can be detected if they are simply near an actual surface. For hypothesized surfels which are within $\pm 30^\circ$ and ± 100 units[3] of and actual surface, the detection rate is nearly 100%. We use the following algorithm to detect surfels.

1. Hypothesize a surfel in world coordinates.
2. Select images from cameras which can image the surfel.
3. Reproject the selected images onto the surfel.
4. Select one of the reprojected regions as a key region to match the others against.
5. For each region:
   (a) Determine the shift which best aligns the region with the key region.
   (b) Estimate color correction which produces the best match with the key region.
   (c) Calculate best match with the key region.
6. Evaluate match set:
   - If good enough $\Rightarrow$ done.
   - If not $\Rightarrow$ goto 4.

Once a surface has been detected, the hypothesized position and orientation can be updated using the geometry of the matching regions and gradient information in the regions respectively. Detected surfels which are not false positives

[3] One unit is 0.1 foot.

typically converge to within 1 unit and a few degrees[4]. The detected surfel's position and orientation are localized using the following algorithm.

1. Until convergence:
   (a) Update the surfel's position.
   (b) Update surfel's orientation.
   (c) Reevaluate the match set.

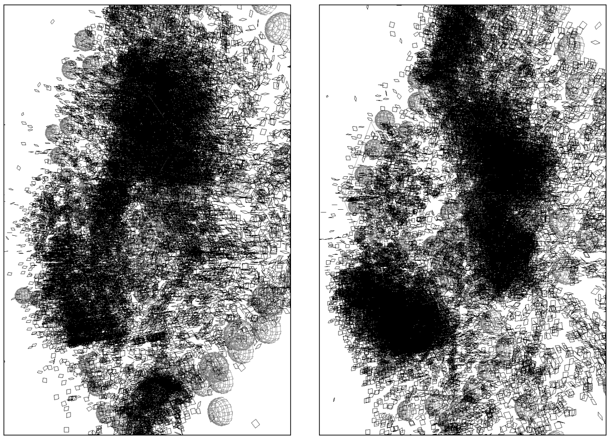


Fig. 10. Raw surfels.

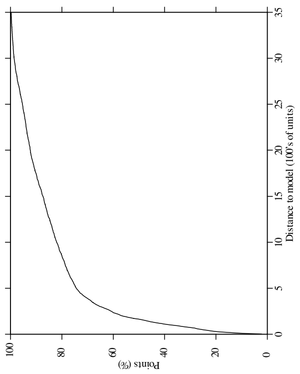


Fig. 11. Distance to nearest model surface.

Figure 10 shows the results of applying the detection and localization algorithms to the dataset described in Section 2. Figure 11 shows the distribution of distances to the nearest model surfaces. Notice that while there are a significant number of false positives, most of the detected surfels are near actual surfaces.

4 Surfels to Surfaces

The results presented in the last section are purely local and make no attempt to reject false positives. This section explores several geometric constraints which together eliminate nearly all false positives.

4.1 Camera Updates

The shifts introduced in the last section are unconstrained. The actual image displacements caused by camera calibration error should be consistent with a translation of the camera center and a rotation about it. To enforce this constraint we use the following algorithm:

[4] For a more complete discussion of detection and localization see [10].

1. For each camera:
   (a) Collect the shifts used to detect surfels.
   (b) Calculate a first-order camera calibration update using these shifts and a linear least-squares solution.
   (c) Use the first-order solution to filter the shifts.
   (d) Calculate the final camera calibration update using non-linear optimization techniques.
   (e) Remove matching regions with shifts that are not consistent with the final update.
2. Prune surfels which no longer meet the match criteria (i.e. too many matching regions have been removed because of inconsistent shifts).

This simple algorithm significantly improves the camera calibration (on average greater than a 3 fold improvement is achieved) and the remaining residuals are consistent with the nonlinear distortion which we have not modeled or corrected for. Figure 12 shows the consistent surfels remaining after applying this algorithm to the raw reconstruction shown in Figure 10 and Figure 13 shows the distribution of distances to the nearest model surface. A number of the consistent surfels come from objects which are not in the reference model. The cluster of surfels between the building outlines near the top center of Figure 12 is one example. These surfels come from a nearby building.

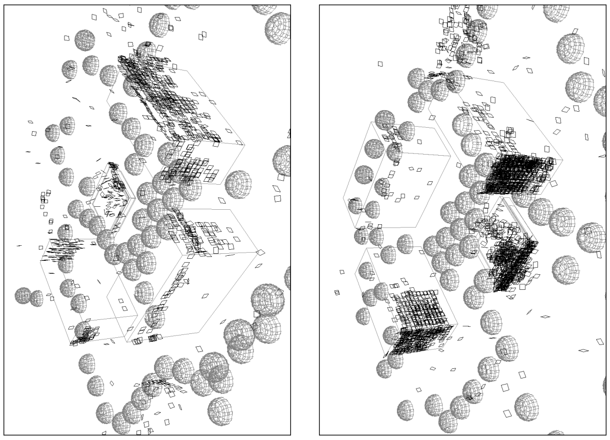


Fig. 12. Consistent surfels.

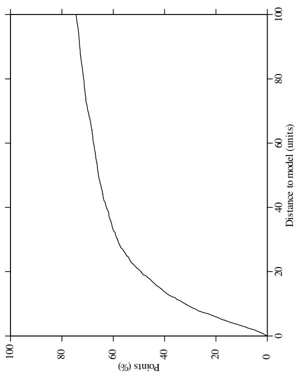


Fig. 13. Distribution of error for consistent surfels.

4.2 One Pixel One Surfel

Each pixel in each image should contribute to at most one surfel. Deciding which surfel is the hard part. Detection and localization as described in the last section do not enforce this constraint and as a result even after enforcing a consistent calibration update there are many image regions which contribute to multiple surfels. We eliminate them in a soft manner using the following algorithm.

1. Score each surfel based on the number of contributing cameras and the number of neighbors.
2. For each surfel S_a.
   (a) For each region which contributes to S_a.
      i. For each surfel S_b with a score higher than S_a, if the region also contributes to S_b.
         A. De-weight the regions contribution to S_a.
   (b) If the match score is no longer sufficient, prune S_a.

A surfel S_a is considered a neighbor of S_b if 1) the distance from the center of S_b to the plane containing S_a (the normal distance); 2) the distance from the projection of the center of S_b onto the plane containing S_a and the center of S_a (the tangential distance); and, 3) the angle between two orientations are all small enough. This notion of neighbors is essentially a smoothness constraint and is also used to group surfels. Figure 14 shows the reconstruction after pruning multiple contributions and Figure 15 shows the distribution of distances to the nearest model surface.

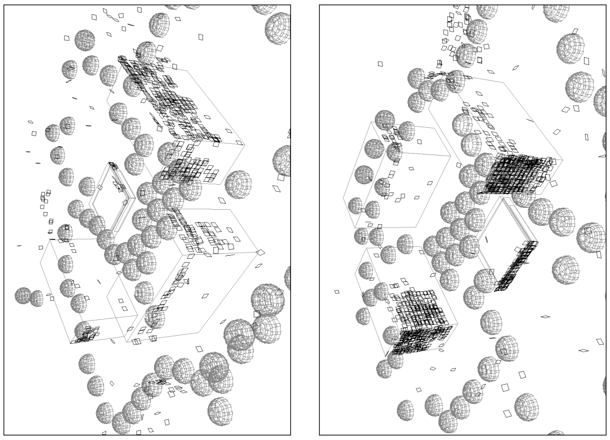


Fig. 14. Surfels after pruning multiple contributions.

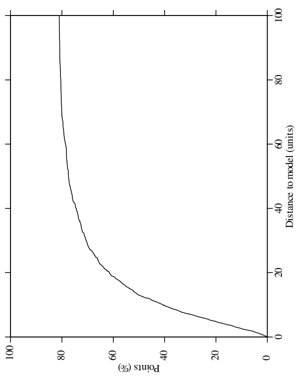


Fig. 15. Distribution of error for pruned surfels.

4.3 Grouping Surfels

The buildings we are trying to model are much larger than an individual surfel. Therefore, a large number of surfels should be reconstructed for each actual surface. Using the notion of neighbors described above, we group the reconstructed surfels as follows:

1. For each surfel S_a.
   (a) For each surfel S_b already assigned a group.

    i. If S_a and S_b are neighbors.
        A. If S_a has not already been assigned to a group, then assign S_a to the group containing S_b.
        B. Otherwise merge the groups containing S_a and S_b.
  (b) If S_a has not been assigned to a group, then create a new group and assign S_a to it.

In practice we retain only groups which have at least a minimum number of (typically four) surfels. All of the surfels in a group should come from the same surface. This notion of grouping places no restriction on the underlying surface other than smoothness (e.g. it may contain compound curves). Figure 16 shows the reconstruction after grouping and removing groups with fewer than four surfels. Nearly all of the surfaces in the reference model have at least one corresponding group. Figure 17 shows the distribution of distances to the nearest model surface.

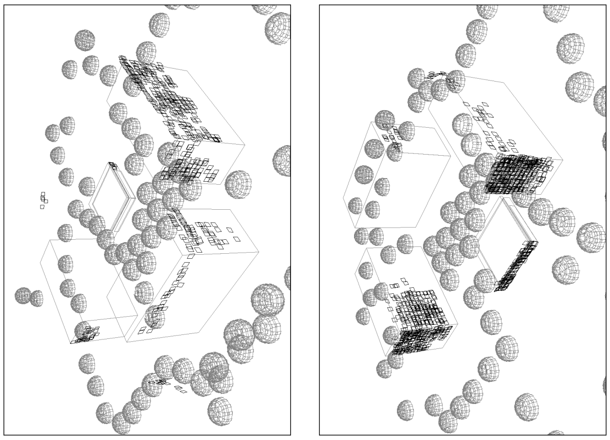


Fig. 16. Surfels after grouping.

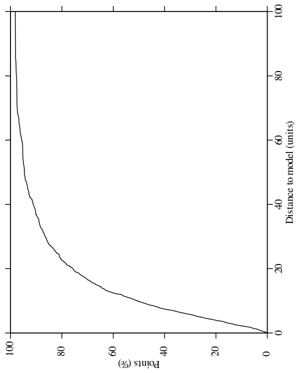


Fig. 17. Distribution of error for grouped surfels.

4.4 Growing Surfaces

Many of the groups shown in figure 16 do not completely cover the underlying surface. There are several reasons why surfels corresponding to actual surfaces might not produce a valid match set. The main one is soft occlusion from tree branches. Another is local maxima encountered while finding the best shifts and updating the surfel's normal. We use the following algorithm to grow surfaces:

1. For each group.
   (a) Create an empty list of hypothesized surfels.
   (b) For each hypothesized surfel.

i. Test using the detection and localization algorithms.
ii. If a match.
A. Add to the group.
B. Test against each surfel in each of the other groups.
If a neighbor, merge the two groups.
(c) Use the next surfel in the group S_a to generate new hypotheses and goto Step 1b.

The hypotheses in Step 1c are generated from S_a by considering the eight nearest neighbors in the plane containing S_a. The shifts and illumination corrections associated with S_a are used as initial values for each hypothesis in Step 1(b)i. Figure 18 shows the reconstruction after growing. After growing, the coverage of each surface is nearly complete. Figure 19 shows the distribution of distances to the nearest model surface.

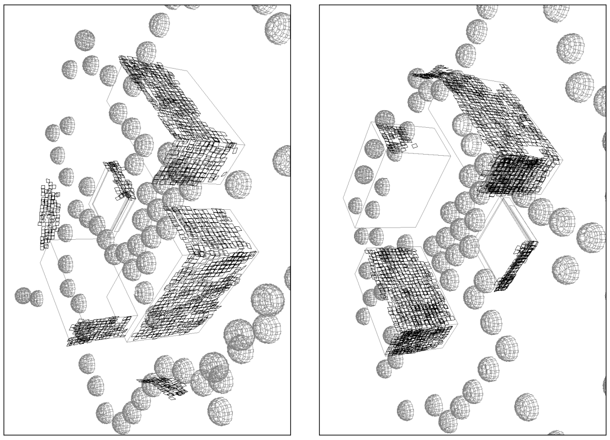
4.5 Extracting Models and Textures

So far, the only assumption we have made about the structure of the world is that locally it can be approximated by a plane. All of the buildings imaged in our dataset are composed of planar faces, therefore we simply fit planes to the groups identified in the previous section. In this case, a face is equivalent to a large surfel. Figure 20 shows the reconstructed faces. A total of 15 surfaces were recovered. Figure 21 shows the distribution of distances to the nearest model surface. As noted previously, many surfels come from structures not in the reference model. Three of the reconstructed surfaces fall into this category, hence Figure 21 has a maximum of 80.

Using the illumination corrections calculated during detection and localization we can transform the images which contribute to a face into a common color space. To obtain the texture associated with each face, we simply reproject the color corrected images and average the results. Figure 22 shows two views of the reconstructed textures. Notice that the rows of window in adjacent faces are properly aligned. This occurs even though no constraints between faces are imposed.

4.6 Discussion

This section uses several simple geometric constraints to remove virtually all false positives from the purely local reconstruction described in Section 3. After imposing consistent calibration updates, removing multiple contributions and grouping, the remaining surfels are excellent seeds for growing surfaces. Of the 16 surfaces in the reference model, 12 were recovered. All of the remaining surfaces are severely occluded by trees. Nearly all of the images are similar to the upper left-hand image of Figures 6 and 7. In spite of this several surfels were recovered on two of these surfaces, however they did not survive the grouping process. In addition to being severely occluded by trees, the other two surfaces have very little texture and one of them suffers from a lack of data. Three surfaces from



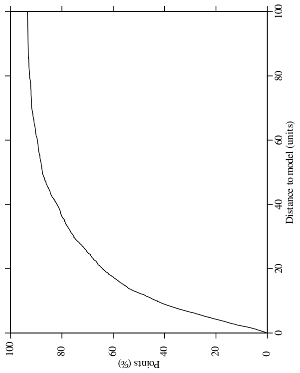
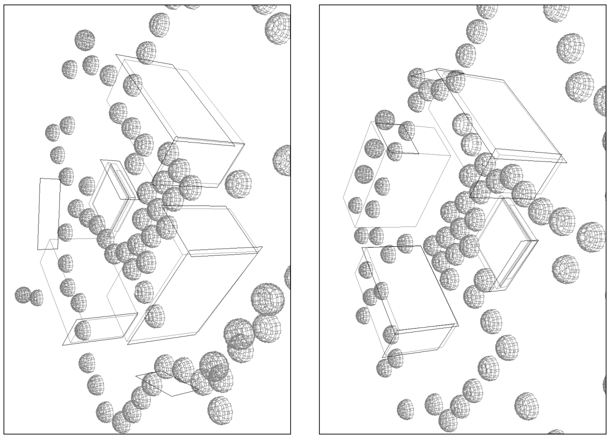


Fig. 18. Surfels after growing.

Fig. 19. Distribution of error for grown surfels.



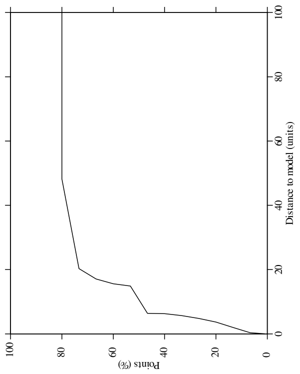


Fig. 20. Raw model surfaces.

Fig. 21. Distribution of error for model surfaces.

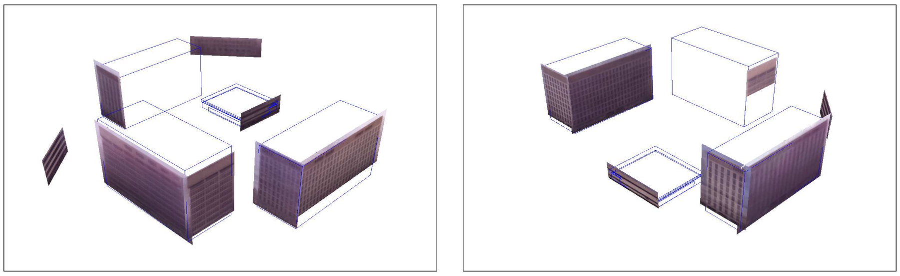


Fig. 22. Textured model surfaces.

adjacent buildings not contained in the model were also recovered. The face near the top center of the upper image in Figure 22 is from the Parson's lab. The surfaces on the left of the upper and the right of the lower image is from Draper lab.

5 Conclusions

This paper presents a novel method for automatically recovering dense surfels using large sets (1000's) of calibrated images taken from arbitrary positions within the scene. Physical instruments, such as Global Positioning System (GPS), inertial sensors, and inclinometers, are used to estimate the position and orientation of each image. Long baseline images improve the accuracy; short baselines and the large number of images simplify the correspondence problem. The initial stage of the algorithm is completely local enabling parallelization and scales linearly with the number of images. Subsequent stages are global in nature, exploit geometric constraints, and scale quadratically with the complexity of the underlying scene [5].

We describe techniques for:

- Detecting and localizing surfels.
- Refining camera calibration estimates and rejecting false positive surfels.
- Grouping surfels into surfaces.
- Growing surfaces along a two-dimensional manifold.
- Producing high quality, textured three-dimensional models from surfaces.

Some of our approach's most important characteristics are:

- It is fully automatic.
- It uses and refines noisy calibration estimates.
- It compensates for large variations in illumination.
- It matches image data directly in three-dimensional space.
- It tolerates significant soft occlusion (e.g. tree branches).
- It associates, at a fundamental level, an estimated normal (eliminating the frontal-planar assumption) and texture with each surfel.

Our algorithms also exploit several geometric constraints inherent in three-dimensional environments and scales well to large sets of images. We believe that these characteristics will be important for systems which automatically recover large-scale high-quality three-dimensional models. A set of about 4000 calibrated images was used to test our algorithms. The results presented demonstrate that they can be used for three-dimensional reconstruction. To our knowledge, the City Scanning project (e.g. [4] and the work presented in this paper) is the first to produce high-quality textured models from such large image sets. The image sets used are nearly two orders of magnitude larger than the largest sets used by other approaches. The approach presented in this paper, recovering dense surfels by matching raw image data directly in three-dimensional space, is unique among the City Scanning approaches.

[5] For a complete discussion of complexity and future work see [10].

References

[1] Shawn Becker and V. Michael Bove, Jr. Semiautomatic 3-D model extraction from uncalibrated 2-D camera views. In *Visual Data Exploration and Analysis*, volume 2410, pages 447–461. SPIE, February 1995. San Jose, CA.

[2] R. Collins, C. Jaynes, F. Stolle, X. Wang, Y. Cheng, A. Hanson, and E. Riseman. A system for automated site model acquisition. In *Integrating Photogrammetric Techniques with Scene Analysis and Machine Vision II*, volume 7617. SPIE, April 1995. Orlando, FL.

[3] Robert T. Collins. A space-sweep approach to true multi-image matching. In *Computer Vision and Pattern Recognition (CVPR '96 Proceedings)*, pages 358–363, June 1996. San Francisco, CA.

[4] Satyan Coorg. *Pose Imagery and Automated 3-D Modeling of Urban Environments*. PhD thesis, MIT, 1998.

[5] Satyan Coorg, Neel Master, and Seth Teller. Acquisition of a large pose-mosaic dataset. In *Computer Vision and Pattern Recognition (CVPR '98 Proceedings)*, pages 872–878, June 1998. Santa Barbara, CA.

[6] Douglas S. J. De Couto. Instrumentation for rapidly acquiring pose-imagery. Master's thesis, MIT, 1998.

[7] Paul E. Debevec, Camillo J. Taylor, and Jitendra Malik. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In *Computer Graphics (SIGGRAPH '96 Proceedings)*, pages 11–20, August 1996.

[8] Kiriakos N. Kutulakos and Steven M. Seitz. A theory of shape by space carving. CS Technical Report 692, University of Rochester, May 1998.

[9] Kiriakos N. Kutulakos and Steven M. Seitz. What do n photographs tell us about 3D shape? CS Technical Report 680, University of Rochester, January 1998.

[10] J.P. Mellor. Automatically recovering geometry and texture from large sets of calibrated images. Technical Report AITR–1674, MIT, October 1999.

[11] J.P. Mellor, Seth Teller, and Tomás Lozano-Pérez. Dense depth maps from epipolar images. In *Image Understanding Workshop (IUW '97 Proceedings)*, volume 2, pages 893–900, May 1997. New Orleans, LA.

[12] J.P. Mellor, Seth Teller, and Tomás Lozano-Pérez. Dense surface patches from thousands of pose images. In *Image Understanding Workshop (IUW '98 Proceedings)*, volume 2, pages 537–542, November 1998. Monterey, CA.

[13] Steven M. Seitz and Charles R. Dyer. Photorealistic scene reconstruction by voxel coloring. In *Computer Vision and Pattern Recognition (CVPR '97 Proceedings)*, pages 1067–1073, 1997. Puerto Rico.

[14] Heung-Yeung Shum, Mei Han, and Richard Szeliski. Interactive construction of 3D models from panoramic mosaics. In *Computer Vision and Pattern Recognition (CVPR '98 Proceedings)*, pages 427–433, June 1998. Santa Barbara, CA.

[15] Richard Szeliski and David Tonnesen. Surface modeling with oriented particle systems. In *Computer Graphics (SIGGRAPH '92 Proceedings)*, volume 26, pages 185–194, July 1992.

[16] Seth Teller. Automated urban model acquisition: Project rationale and status. In *Image Understanding Workshop (IUW '98 Proceedings)*, volume 2, pages 455–462, November 1998. Monterey, CA.

[17] Seth Teller. Toward urban model acquisition from geo-located images. In *Pacific Graphics '98*, 1998. Singapore.