Evaluation of Large Scale Scene Reconstruction

Paul Merrell, Philippos Mordohai, Jan-Michael Frahm, and Marc Pollefeys
Department of Computer Science
University of North Carolina, Chapel Hill, USA

Abstract

We present an evaluation methodology and data for large scale video-based 3D reconstruction. We evaluate the effects of several parameters and draw conclusions that can be useful for practical systems operating in uncontrolled environments Unlike the benchmark datasets used for the binocular stereo and multi-view reconstruction evaluations, which were collected under well-controlled conditions, our datasets are captured outdoors using video cameras mounted on a moving vehicle. As a result, the videos are much more realistic and include phenomena such as exposure changes from viewing both bright and dim surfaces, objects at varying distances from the camera, and objects of varying size and degrees of texture. The dataset includes ground truth models and precise camera pose information. We also present an evaluation methodology applicable to reconstructions of large scale environments. We evaluate the accuracy and completeness of reconstructions obtained by two fast, visibility-based depth map fusion algorithms as parameters vary.

1. Introduction

The problem of 3D reconstruction from video is an important topic in computer vision. Recently, the acquisition of videos primarily of cities using cameras mounted on vehicles has been the focus of several research and commercial organizations. The goal is to generate high-quality 3D models of complete cities to be used primarily for visualization. In this paper we propose a system that evaluates two properties of 3D reconstruction: geometric accuracy and completeness. The metrics are similar to those of the Multi-view Stereo Evaluation website (http://vision.middlebury.edu/mview/) [14], but they are used for large scale scene reconstruction and not for the reconstruction of a single object. Both properties directly contribute to the visual quality of the reconstruction, since inaccuracies in geometry create in disturbing artifacts as the user changes viewpoint, while low completeness coverage also reduces the effectiveness of the visualization. Besides the domain, the key difference between our work and previous evaluation efforts [13, 14] is that our dataset is more representative of what would be expected in a real-world application. Our dataset depicts objects of varying sizes and texture properties such as specular reflections while the distance from the camera to the scene and the brightness vary.

Our initial test dataset consists of approximately 3,000 frames of video of a large building captured by cameras mounted on a vehicle equipped with a Global Positioning System (GPS) and an Inertial Navigation System (INS). We use, as baseline reconstruction algorithms, the two visibility-based depth map fusion algorithms presented in [10]. They operate like a sliding window on a set of potentially noisy depth maps and their output is a smaller set of depth maps that minimize visibility violations, improve the accuracy of the depth estimates and reduce the redundancy of the input depth maps which have large overlaps with each other. The basic concepts of the algorithms are presented in Section 4, while more details can be found in [10].

In order for our evaluation software to be independent of the algorithm being evaluated, we adopted an approach that evaluates 3D points and thus applies to point-based and mesh-based representations. For meshes, the vertices of the mesh are evaluated. In Section 5, we evaluate the effects of parameters, such as the window size used in stereo matching, the number of depth estimates per pixel, the number of images used for the computation of each depth map and the number of depth maps that are fused to produce one fused depth map, on both accuracy and completeness.

2. Related Work

In this section, we review 3D reconstruction methods that are applicable to large scale scenes. Our focus is mainly on methods that use image and video inputs, but we also review important approaches for general range data. Surveys of binocular and multiple-view stereo research, which can be components in a large scale reconstruction system, but are not able to process more than a few dozen images, can be found in [4, 15, 13, 1, 14].

A widely used algorithm for merging two triangular meshes was proposed by Turk and Levoy [16]. A volumetric approach that also received a lot of attention was later presented by Curless and Levoy [3] who compute a cumulative weighted distance function from the depth estimates. The surface is then extracted as the zero-level set of the distance function. Wheeler et al. [17] increase the robustness of [3] by requiring a minimum amount of support for each depth estimate before it is integrated in the final model.

Koch et al. [8] presented a video-based reconstruction approach which detects binocular pixel correspondences and then links depth estimates across more cameras to refine their position and reduce uncertainty. Narayanan et al. [11] compute depth maps using multi-baseline stereo and merge them to produce viewpoint-based visible surface models. Koch et al. [9] developed a volumetric approach in which estimates vote for voxels in a probabilistic way. Sato et al. [12] also proposed a volumetric method based on voting. Each depth estimate votes not only for likely surfaces but also for free space between the camera and the surfaces. Goesele et al. [6] use a two-stage algorithm which merges depth maps produced by a simple multiple-view stereo module. Depth estimates are rejected if cross-correlation is not large enough for at least two target views. The remaining depth estimates are merged using [3].

Our depth map fusion approach [10] is viewpoint-based and generates consensus depths by taking visibility constraints into account. We present two algorithms that operate on the set of depth hypotheses for each pixel of a reference view to produce one reliable depth per pixel. The evaluation metrics we use resemble those of Seitz et al. [14]. Their evaluation is performed using ground truth models of two objects that have been accurately reconstructed using active sensors. Accuracy is defined as the distance d such that 90% of the reconstructed points are within d of the ground truth. Completeness is defined as the percentage of points on the ground truth model that are within $1.25mm$ of the reconstruction.

3. Evaluation Methodology

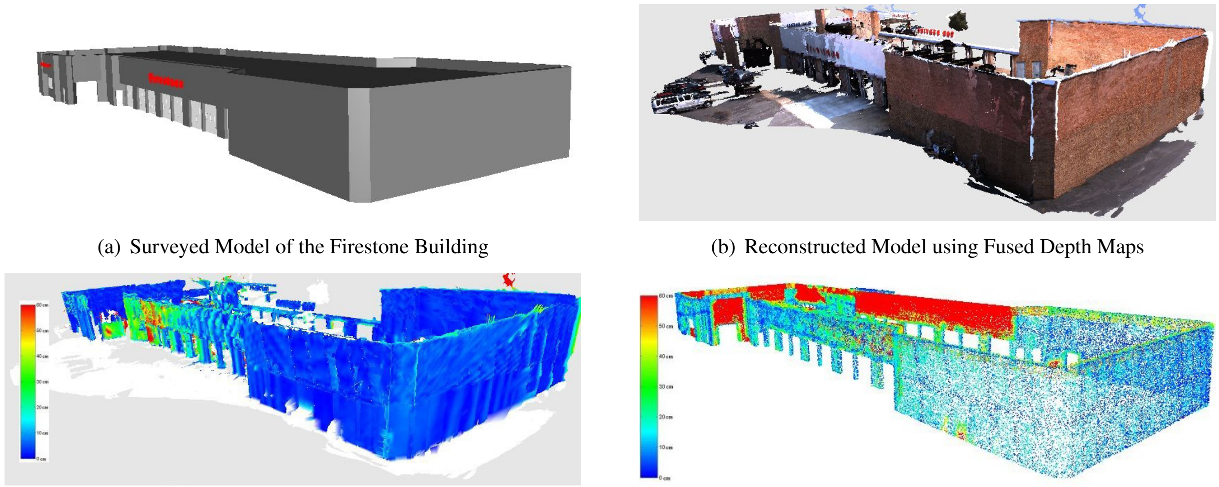
In this section, we describe the ground truth data, the input data that are available to the algorithms, and the evaluation procedure. The ground truth model used for the results presented in this paper is a Firestone building. Its dimensions are $80 \times 40m$ and it was surveyed at an accuracy of $6mm$. The surveyed model can be seen in Fig. 1(a). There are several objects such as parked cars that were not surveyed even though they appear in the video. Several of the doors of the building were closed during the collection of the ground truth data, but were left open during the collection of the video data. This caused some of the interior of the building to be reconstructed. The ground also was not surveyed. Since accurate measurements of all of these reconstructed objects were unavailable they are manually removed from the evaluation.

The test dataset includes 3,000 video frames of the exterior of the Firestone building captured by two cameras on a vehicle driving around the building. Using GPS and inertial measurements, the geo-registered position and orientation of the vehicle is available at each frame. The position and orientation of the cameras is calibrated with respect to the GPS unit on the vehicle. Given this information, the camera poses can be computed. There is virtually no overlap between the two cameras, since one of them was pointed horizontally towards the middle and bottom of the building and the other was pointed up $30°$. The input to the reconstruction algorithm is a set of frames with known camera intrinsic parameters and poses. Since there is no overlap between the cameras, reconstruction can only be done using a single-camera video-based method. The data is representative of the kind of data expected in a real-world application where there are many uncontrolled variables such as variations in texture, brightness, and the distance between the camera and the scene.

We evaluate two quantities of the reconstructed model: accuracy and completeness. To measure the accuracy of each reconstructed vertex, the distance from the vertex to the nearest triangle of the ground truth model is calculated. The error measurements for each part of a reconstruction using the confidence-based algorithm described in the next section are displayed in Fig. 1(c). Completeness measures how much of the building was reconstructed. Sample points are chosen at random on the surface of the ground truth model so that there are on average 50 sample points per square meter of surface. The distance from each sample point to the nearest reconstructed vertex is measured. A visualization of these distances are shown for one of the reconstructions in Figure 1(d). Unless otherwise specified, a threshold of $50cm$ is used for the completeness measures presented in Section 5.

4. Baseline depth map fusion algorithms

In this section, we briefly describe the two visibility-based depth map fusion algorithms of [10]. We employ a two-stage strategy that allows us to achieve very high processing speeds. By decoupling the problem into the reconstruction of depth maps from sets of images followed by the fusion of these depth maps, we are able to use simple fast algorithms that can be easily ported to the GPU. Conflicts and errors in the depth maps are identified and resolved in the fusion stage. In this step, a set of N depth maps from neighboring camera positions are combined into a single depth map for one of the views. The end result is a fused depth map from the perspective of one of the original viewpoints. This viewpoint is called the reference view and is typically selected to be the center viewpoint of the set. Processing is performed in a sliding window of the data, since it is impossible to process thousands of images simultaneously. In



(a) Surveyed Model of the Firestone Building

(b) Reconstructed Model using Fused Depth Maps

(c) Visualization of the accuracy evaluation, where white indicates parts of the model that have not been surveyed and blue, green and red indicate errors of $0cm$, $30cm$ and $60cm$ or above, respectively. Please view on a color display.

(d) Completeness of the Firestone building. The color coding is the same as in (c). Red areas mostly correspond to unobserved or untextured areas.

Figure 1. Firestone Building Accuracy and Completeness Evaluation

addition, the fusion step produces a more compact representation of the data because the number of fused depth maps is a fraction of the original number of depth maps. Much of the information in the original depth maps is redundant since many of the closely-spaced viewpoints observe the same surface. After fusion, a mesh is generated using a quadtree approach that minimizes the number of triangles maintaining geometric accuracy. See [10] for details. The same module detects overlaps between consecutive fused depth maps and merges the overlapping surfaces.

4.1. Multiple-View Stereo

The first stage of the reconstruction computes depth maps from sets of images captured from a single moving camera with known poses using plane-sweeping stereo [2, 18, 5]. The depth map is computed for the central image in a set of 5 to 11 images. At each pixel, several depth hypotheses are tested in the form of planes. For each plane, the depth for a pixel is computed by intersecting the ray emanating from the pixel with the hypothesized plane. All images are projected onto the plane and a cost for the hypothesized depth is calculated as the sum of absolute intensity differences (SAD). The set of images is divided in two halves, one preceding and one following the reference image. The SAD between each half of the images and the reference view is calculated in square windows. The minimum of the two sums is the cost of the depth hypothesis [7]. This scheme is effective against occlusions, since in general the visibility of a pixel does not change more than once in an image sequence. The depth of each pixel is estimated to be the depth d_0 with the lowest cost. Each pixel is processed independently allowing non-planar surfaces to be reconstructed.

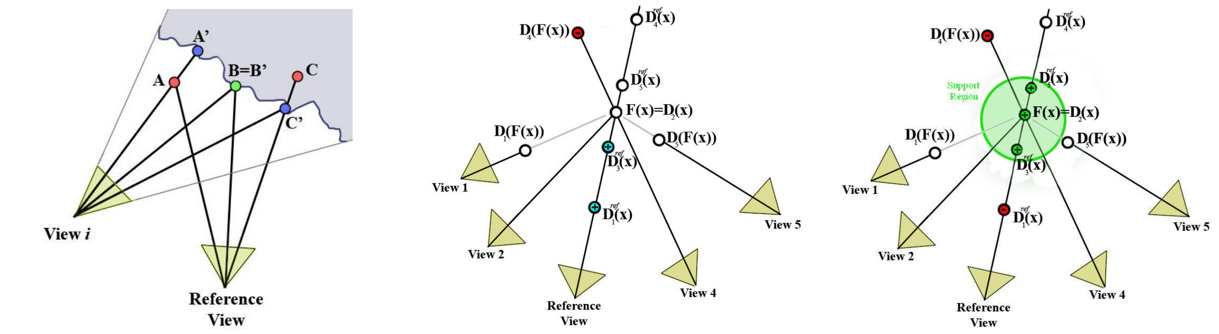
The depth with the lowest cost may not be the true depth due to noise, occlusion, lack of texture, surfaces that are not aligned with the plane direction, and many other factors. Thus, a measure of confidence of each depth estimate is important. Let $c(\mathbf{x}, d)$ be the matching cost for depth d at pixel $\mathbf{x}$. We wish to estimate the likelihood that the true depth, d_o, does not have the lowest cost after the cost is perturbed. Assuming Gaussian noise, this likelihood is proportional to: $e^{-(c(\mathbf{x},d)-c(\mathbf{x},d_0))^2/\sigma^2}$ for some σ that depends on the strength of the noise. The confidence $C(\mathbf{x})$ is defined as the inverse of the sum of these probabilities for all possible depths:

$$C(\mathbf{x}) = \left(\sum_{d \neq d_0} e^{-(c(\mathbf{x},d)-c(\mathbf{x},d_0))^2/\sigma^2} \right)^{-1} \qquad (1)$$

This equation produces a high confidence when the cost has a single sharp minimum. The confidence is low when the cost has a shallow minimum or several low minima.

4.2. Visibility-Based Fusion

The input to the fusion step is a set of N depth maps denoted by $D_1(\mathbf{x}), D_2(\mathbf{x}), \ldots, D_N(\mathbf{x})$ which record the estimated depth of each pixel of the N images. Each depth map has an associated confidence map labeled $C_1(\mathbf{x}), C_2(\mathbf{x}), \ldots, C_N(\mathbf{x})$ computed according to (1). Typically the central viewpoint is selected as the reference view and we seek a depth estimate for each of its pixels. The current estimate of the 3D point seen at pixel $\mathbf{x}$ of the reference



(a) Visibility relations between points (b) Stability calculation (c) Support estimation

Figure 2. (a) Visibility relations between points. The point A' seen in view i has its free space violated by A seen in the reference view. B' supports B. C seen in the reference view is occluded by C'. (b) Stability Calculation. In this example, there are two occlusions which raise stability and one free-space violations which lowers it. The stability is +1. (c) Support calculation. Three measurements are close to the current estimate and add support to it. Outside the support region, there is one occlusion and one free-space violation which lower the support.

view is called $\hat{F}(\mathbf{x})$. $R_i(\mathbf{X})$ is the distance between the center of projection of viewpoint i and the 3D point $\mathbf{X}$. To simplify the notation, we define the term $\hat{f}(\mathbf{x}) \equiv R_{ref}(\hat{F}(\mathbf{x}))$ which is the distance of the current depth estimate $\hat{F}(\mathbf{x})$ for the reference camera.

The first step of fusion is to render each depth map into the reference view. When multiple depth values project onto the same pixel, the nearest depth is kept. Let D_i^{ref} be the depth map D_i rendered into the reference view and C_i^{ref} be the confidence map rendered in the reference view. Given a 3D point $\mathbf{X}$, we need a notation to describe the value of the depth map D_i at the location where $\mathbf{X}$ projects into view i. Let $P_i(\mathbf{X})$ be the image coordinates of the 3D point $\mathbf{X}$ projected into view i. To simplify the notation, the following definition is used $D_i(\mathbf{X}) \equiv D_i(P_i(\mathbf{X}))$. $D_i(\mathbf{X})$ is likely to be different from $R_i(\mathbf{X})$ which is the distance between $\mathbf{X}$ and the camera center.

Our approach considers three types of visibility relationships between hypothesized depths in the reference view and computed depths in the other views. These relations are illustrated in Fig. 2(a). The point A' observed in view i is behind the point A observed in the reference view. There is a conflict between the measurement and the hypothesized depth since view i would not be able to observe A' if there truly was a surface at A. We say that A violates the free space of A'. This occurs when $R_i(A) < D_i(A)$.

In Fig. 2(a), B' is in agreement with B since they are in the same location. In practice, we define points B and B' as being in agreement when $\frac{|R_{ref}(B)-R_{ref}(B')|}{R_{ref}(B)} < \epsilon$.

The point C' observed in view i is in front of the point C observed in the reference view. There is a conflict between these two measurements since it would be impossible to observe C if there truly was a surface at C'. We say that C' occludes C. This occurs when $D_i^{ref}(\mathbf{x}) < \hat{f}(\mathbf{x}) = D_{ref}(\mathbf{x})$.

Note that operations for a pixel are not performed on a single ray, but on rays from all cameras. Occlusions are defined on the rays of the reference view, but free space violations are defined on the rays of the other depth maps. The reverse depth relations (such as A behind A' or C in front of C') do not represent visibility conflicts.

The raw stereo depth maps give different estimates of the depth at a given pixel in the reference view. We first present a method that tests each of these estimates and selects the most likely candidate by exhaustively considering all occlusions and free-space constraints. We then present an alternative approach that selects a likely candidate upfront based on the confidence and then verifies that this estimate agrees with most of the remaining data. The type of computations required in both approaches are quite similar. Most of the computation time is spent rendering a depth map seen in one viewpoint into another viewpoint. These computations can be performed efficiently on the GPU.

4.3. Algorithm 1: Stability-Based Fusion

If a depth map occludes a depth hypothesis $\hat{F}(\mathbf{x})$, this indicates that the hypothesis is too far away from the reference view. If the current depth hypothesis violates a free-space constraint, this indicates the hypothesis is too close to the reference view. The stability of a point $S(\mathbf{x})$ is defined as the number of depth maps that occlude $\hat{F}(\mathbf{x})$ minus the number of free-space violations. Stability measures the balance between these two types of visibility violations. A point is stable if the stability is greater than or equal to zero. If the stability is negative, then most of the depth maps indicate that $\hat{F}(\mathbf{x})$ is too close to the camera to be correct. If the stability is positive then at least half of the depth maps indicate that $\hat{F}(\mathbf{x})$ is far enough away from the reference camera. Stability generally increases as the point moves further away from the camera. The final fused depth is selected to be the closest depth to the camera for which stability is non-negative. This depth is not the median depth along the viewing ray since free-space violations are defined on rays that do not come from the reference view. This depth is balanced in the sense that the amount of evidence that indicates it is too close is equal to the amount of evidence that indicates it is too far away.

With this goal in mind, we construct an algorithm to find the closest stable depth. To begin, all of the depth maps are rendered into the reference view. In the example of Fig. 2(b), five depth maps are rendered into the reference view. The closest depth is selected as the initial estimate. In the example, the closest depth is $D_1^{ref}(\mathbf{x})$ and so its stability is evaluated first. The point is tested against each depth map to determine if the depth map occludes it or if it violates the depth map's free space. If the depth estimate is found to be unstable, we move onto the next closest depth. Since there are N possible choices, the proper depth estimate is guaranteed to be found after $N-1$ iterations. The total number of depth map renderings is bound by $O(N^2)$. In the example, the closest two depths $D_1^{ref}(\mathbf{x})$ and $D_3^{ref}(\mathbf{x})$ were tested first. Figure 2(b) shows the test being performed on the third closest depth $D_2^{ref}(\mathbf{x})$. A free-space violation and two occlusions are found and thus the stability is positive. In this example, $D_2^{ref}(\mathbf{x})$ is the closest stable depth.

The final step is to compute a confidence value for the estimated depth. Each depth estimate $D_i(\hat{F}(\mathbf{x}))$ for the pixel and the selected depth $R_i(\hat{F}(\mathbf{x}))$ are compared. If they are within ϵ, the depth map supports the final estimate. The confidences of all the estimates that support the selected estimate are added. The resulting fused confidence map is passed on to the mesh construction module.

4.4. Algorithm 2: Confidence-Based Fusion

Stability-based fusion tests up to $N-1$ different depth hypotheses. In practice, most of these depth hypotheses are close to one another, since the true surface is likely to be visible and correctly reconstructed in several depth maps. Instead of testing so many depth estimates, an alternative approach is to combine multiple close depth estimates into a single estimate and then perform only one test. Because there is only one hypothesis to test, there are only $O(N)$ renderings to compute. This approach is typically faster than stability-based fusion which tests $N-1$ hypotheses and computes $O(N^2)$ renderings, but the early commitment may cause additional errors.

Combining Consistent Estimates Confidence-based fusion also begins by rendering all the depth maps into the reference view. The depth estimate with the highest confidence is selected as the initial estimate for each pixel. At each pixel $\mathbf{x}$, we keep track of two quantities which are updated iteratively: the current depth estimate and its level of support. Let $\hat{f}_0(\mathbf{x})$ and $\hat{C}_0(\mathbf{x})$ be the initial depth estimate and its confidence value. $\hat{f}_k(\mathbf{x})$ and $\hat{C}_k(\mathbf{x})$ are the depth estimate and its support at iteration k, while $\hat{F}(\mathbf{x})$ is the corresponding 3D point.

If another depth map $D_i^{ref}(\mathbf{x})$ produces a depth estimate within ϵ of the initial depth estimate $\hat{f}_0(\mathbf{x})$, it is very likely that the two viewpoints have correctly reconstructed the same surface. In the example of Fig. 2(c), the estimates $D_3(\hat{F}(\mathbf{x}))$ and $D_5(\hat{F}(\mathbf{x}))$ are close to the initial estimate. These close observations are averaged into a single estimate. Each observation is weighted by its confidence according to the following equations:

$$\hat{f}_{k+1}(\mathbf{x}) = \frac{\hat{f}_k(\mathbf{x})\hat{C}_k(\mathbf{x}) + D_i^{ref}(\mathbf{x})C_i(\mathbf{x})}{\hat{C}_k(\mathbf{x}) + C_i(\mathbf{x})} \tag{2}$$

$$\hat{C}_{k+1}(\mathbf{x}) = \hat{C}_k(\mathbf{x}) + C_i(\mathbf{x}) \tag{3}$$

The result is a combined depth estimate $\hat{f}_k(\mathbf{x})$ at each pixel of the reference image and a support level $\hat{C}_k(\mathbf{x})$ measuring how well the depth maps agree with the depth estimate. The next step is to find how many of the depth maps contradict $\hat{f}_k(\mathbf{x})$ in order to verify its correctness.

Conflict Detection The total amount of support for each depth estimate must be above the threshold C_{thres} or else it is discarded as an outlier and is not processed any further. The remaining points are checked using visibility constraints. Figure 2(c) shows that $D_1(\hat{F}(\mathbf{x}))$ and $D_3(\hat{F}(\mathbf{x}))$ occlude $\hat{F}(\mathbf{x})$. However, $D_3(\hat{F}(\mathbf{x}))$ is close enough (within ϵ) to $\hat{F}(\mathbf{x})$ to be within its support region and so this occlusion does not count against the current estimate. $D_1(\hat{F}(\mathbf{x}))$ occludes $\hat{F}(\mathbf{x})$ outside the support region and thus contradicts the current estimate. When such an occlusion takes place the support of the current estimate is decreased by:

$$\hat{C}_{k+1}(\mathbf{x}) = \hat{C}_k(\mathbf{x}) - C_i^{ref}(\mathbf{x}) \tag{4}$$

When a free-space violation occurs outside the support region, as shown with the depth $D_4(\hat{F}(\mathbf{x}))$ in Fig. 2(c), the confidence of the conflicting depth estimate is subtracted from the support according to:

$$\hat{C}_{k+1}(\mathbf{x}) = \hat{C}_k(\mathbf{x}) - C_i(P_i(\hat{F}(\mathbf{x}))) \tag{5}$$

We have now added the confidence of all the depth maps that support the current depth estimate and subtracted the confidence of all those that contradict it. If the support is positive, the majority of the evidence supports the depth estimate and it is kept. If the support is negative, the depth estimate is discarded as an outlier. The fused depth map at this stage contains estimates with high confidence and holes where the estimates have been rejected.

Hole filling After discarding the outliers, there are holes in the fused depth map. In practice, the depth maps of most real-world scenes are piecewise smooth and we assume that any small missing parts of the depth map are most likely to have a depth close to their neighbors. To fill in the gaps, we find all inliers within a $w \times w$ window centered at the pixel we wish to estimate. If there are enough inliers to make a good estimate, we assign the median of the inliers as the depth of the pixel. If there are only a few neighboring inliers, the depth map is left blank. Essentially, this is a median filter that ignores the outliers. In the final step, a median filter with a smaller window w_s is used to smooth out the inliers.

	Median (cm)	Mean (cm)	Completeness
5 Images	3.02	13.02	55%
7 Images	2.78	12.45	55%
11 Images	2.66	12.95	55%

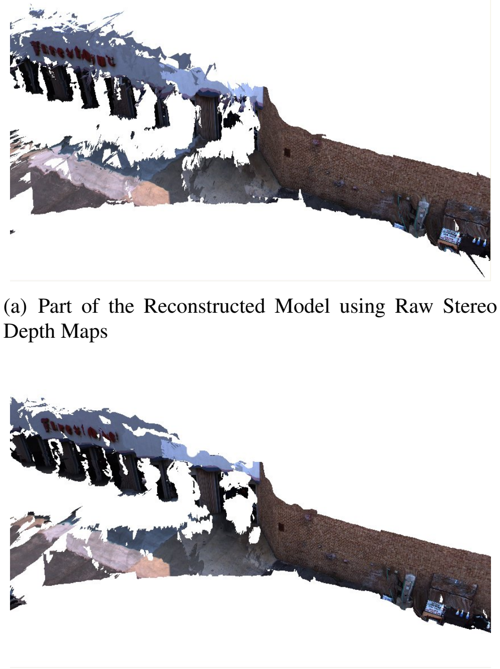
Table 1. Accuracy and Completeness of the Raw Stereo Depth Maps using different numbers of stereo images (single camera).

Stereo Window	4×4	8×8	16×16	32×32
Median Error(cm)	3.40	2.83	2.78	3.17
Mean Error(cm)	13.89	12.76	12.45	11.61
Completeness	39%	48%	55%	55%

Table 2. Accuracy and Completeness of the Raw Stereo Depth Maps using different stereo window sizes (single camera).

5. Results

We tested our methods on the Firestone building and generated reconstructions with different settings of four parameters. The first parameter is the number of images used to compute each stereo depth map using plane sweeping. The second is the number of depth maps that are fused. The third and fourth parameters are the number of planes used during plane sweeping and the size of the window over which the best cost is calculated in stereo. The default values for each of these parameters is 7 images for stereo, using 48 planes and a 16×16 window, and 11 raw depth maps for each fusion step. In the tables that follow, these parameters are set to their default values unless noted otherwise. Every 16 frames, the raw depth maps are fused. The remaining parameters are set to the following values: $\epsilon = 0.05, \sigma = 120, w = 8$ pixels, $w_s = 4$ pixels, and $C_{thres} = 5$. The image size is 256×192. Virtually indistinguishable results are obtained on 512×384 inputs. Unless noted otherwise, we only used the horizontal camera. Using the default settings, confidence-based fusion takes $38ms$ and stability-based fusion takes $51ms$ on a high-end commodity GPU. As the number of depth maps increases, the execution times of stability-based fusion grow quadratically, while those of confidence-based fusion grow linearly.


(a) Part of the Reconstructed Model using Raw Stereo Depth Maps

(b) Part of the Reconstructed Model using Confidence-Based Fusion

Figure 3. Firestone Building Reconstruction

To begin, we evaluated the raw stereo depth maps before any fusion was performed. For the results in Table 3 labeled as *stereo-reference*, we evaluate the raw depth maps from each of the reference views. For the results labeled *stereo-exhaustive*, we evaluated the depth maps from *all* images as the representation of the scene. Tables 1 and 2 show result for non-exhaustive raw stereo. Using more images per stereo computation improves the accuracy without changing the completeness (Table 1). A stereo window of 16×16 pixels gives both the lowest median error and the highest completeness (Table 2). Small stereo windows create noisy depth maps, but large stereo windows may oversmooth the depths. A comparison of the two fusion methods and the raw stereo depth maps is shown in Table 3 and Figs. 4 and 5. Confidence-based fusion reconstructs more of the building, but is less accurate due to the smoothing step and the non-exhaustive search. A close up of the reconstruction is shown using raw stereo (Fig. 3(a)) and using confidence-based fusion (Fig. 3(b)), which reduces the surface noise.

We then compared the results of the two stereo and the two fusion options using frames from *both* cameras. Both fusion methods increase the accuracy while slightly decreasing the completeness of raw stereo. The mean error decreases much more than the median error, since fusion removes gross outliers. Another benefit of fusion is it produces a more compact representation of the data. The models of the Firestone building created by exhaustively using all of the raw stereo depth maps are huge. They contain

Fusion Method	Stereo-exhaustive)	Stereo-reference)	Confidence	Stability
Median Error(cm)	4.87	4.19	2.60	2.19
Mean Error(cm)	40.61	39.20	6.60	4.79
Completeness	94%	83%	73%	66%

Table 3. Accuracy and Completeness for different fusion methods using the default parameters (both cameras).

	Confidence-Based			Stability-Based		
Median Error (cm)	7 Depth Maps	11 Maps	17 Maps	7 Depth Maps	11 Maps	17 Maps
5 Images	2.50	2.35	2.25	2.05	2.00	1.86
7 Images	2.38	2.33	2.26	2.10	2.06	2.01
11 Images	2.36	2.35	2.38	1.36	2.28	2.29

Table 4. Median errors for both fusion methods using different numbers of stereo images and depth maps (single camera).

over 5,600,000 vertices. The models created after fusion are relatively small. The number of vertices is reduced to less than 260,000 and 320,000 vertices for stability-based and confidence-based fusion respectively.

There is a trade off between the accuracy and the completeness of the reconstruction. Some parts of the building are more difficult to accurately reconstruct than others. Areas with little texture are particularly difficult. If the parameters are set so that more of the difficult parts are reconstructed the completeness measure increases, but the accuracy decreases. The trade-off between accuracy and completeness cannot be determined without taking into consideration the objective of each reconstruction system. Accuracy is more critical for certain applications and completeness for others. Accuracy may be more important for applications such as path planning and obstacle avoidance and completeness for image-based rendering. A diagram showing these distances on the model is provided in Fig. 1(c). A histogram of these distances is shown in Fig. 4. Using confidence-based fusion 83% of reconstructed the points were within 5 cm of the ground truth model.

The results in Tables 4 and 5 show that the accuracy tends to improve as the number of depth maps that are fused increases. The number of stereo images in stability-based fusion clearly raises the completeness (Table 6), but lowers the accuracy (Tables 4 and 5). Increasing the size of the stereo window generally increases completeness, but lowers the accuracy (Tables 7 and 8). Confidence-based fusion is better able to deal with small stereo windows. As the number of planes used in plane sweeping increases, the accuracy improves while the amount of completeness hardly changes (Table 9).

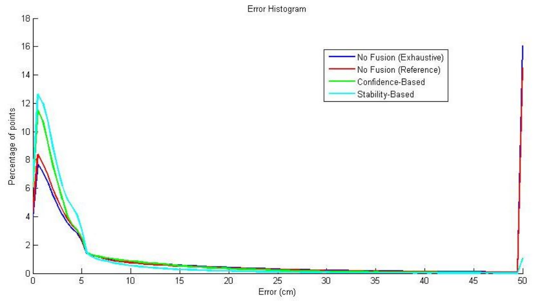


Figure 4. Histogram of Errors in the Firestone Reconstruction (both cameras)

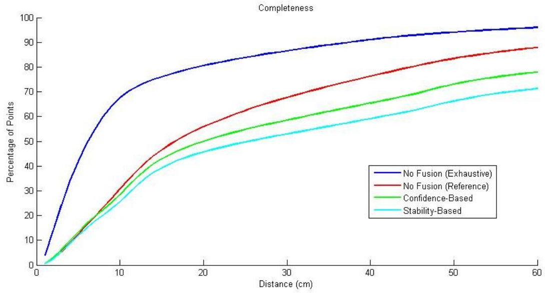


Figure 5. Completeness Measurements. Sample points within a given distance from the reconstruction (both cameras).

Stereo Window	4 × 4	8 × 8	16 × 16	32 × 32
Median Error(cm)	1.96	2.19	2.33	2.67
Mean Error(cm)	4.68	5.46	6.60	8.26
Completeness	40%	47%	50%	50%

Table 7. Accuracy and Completeness for confidence-based fusion using different stereo window sizes (single camera).

6. Conclusion

We have presented a dataset, which we intend to extend by adding more scenes, that contains ground truth for a large building, videos of the building and accurate poses for the cameras at each frame. We also presented an evaluation methodology applicable to reconstructions of large scale environments. The key difference with previous vision-based reconstruction evaluation efforts is the fact that our videos are much more realistic, since they were captured under sunlight at varying distances from the building which in-

	Confidence-Based			Stability-Based		
Mean Error (cm)	7 Depth Maps	11 Maps	17 Maps	7 Depth Maps	11 Maps	17 Maps
5 Images	7.02	6.58	6.30	3.78	3.57	3.46
7 Images	6.99	6.60	6.28	4.49	4.29	4.24
11 Images	6.91	6.74	6.76	5.47	5.32	5.22

Table 5. Mean errors for both fusion methods using different numbers of stereo images and depth maps (single camera).

	Confidence-Based			Stability-Based		
Completeness	7 Depth Maps	11 Maps	17 Maps	7 Depth Maps	11 Maps	17 Maps
5 Images	51%	49%	46%	39%	39%	39%
7 Images	52%	50%	49%	45%	44%	44%
11 Images	52%	51%	50%	51%	51%	51%

Table 6. Completeness Measurements for both fusion methods using different numbers of stereo images and depth maps.

cludes surfaces with very different properties. We hope that our work offers useful insights to research efforts on 3D reconstruction of outdoors scenes.

Stereo Window	4×4	8×8	16×16	32×32
Median Error(cm)	4.44	1.78	2.06	2.82
Mean Error(cm)	6.06	2.83	4.29	8.06
Completeness	8%	33%	44%	51%

Table 8. Accuracy and Completeness for stability-based fusion using different stereo window sizes.

	48 planes	100 planes	150 planes
Median Error(cm)	2.33	2.18	2.16
Mean Error(cm)	6.60	6.37	6.16
Completeness	50%	49%	49%

Table 9. Accuracy and Completeness for Confidence-Based Fusion using different number of planes during plane sweeping.

References

[1] M. Brown, D. Burschka, and G. Hager. Advances in computational stereo. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25(8):993–1008, August 2003.

[2] R. Collins. A space-sweep approach to true multi-image matching. In *Int. Conf. on Computer Vision and Pattern Recognition*, pages 358–363, 1996.

[3] B. Curless and M. Levoy. A volumetric method for building complex models from range images. *SIGGRAPH*, 30:303–312, 1996.

[4] C. Dyer. Volumetric scene reconstruction from multiple views. In L. Davis, editor, *Foundations of Image Understanding*, pages 469–489. Kluwer, Dordrecht, 2001.

[5] D. Gallup, J.-M. Frahm, P. Mordohai, Q. Yang, and M. Pollefeys. Real-time plane-sweeping stereo with multiple sweeping directions. In *Int. Conf. on Computer Vision and Pattern Recognition*, 2007.

[6] M. Goesele, B. Curless, and S. Seitz. Multi-view stereo revisited. In *Int. Conf. on Computer Vision and Pattern Recognition*, pages II: 2402–2409, 2006.

[7] S. Kang, R. Szeliski, and J. Chai. Handling occlusions in dense multi-view stereo. In *Int. Conf. on Computer Vision and Pattern Recognition*, pages I:103–110, 2001.

[8] R. Koch, M. Pollefeys, and L. Van Gool. Multi viewpoint stereo from uncalibrated video sequences. In *European Conf. on Computer Vision*, pages I: 55–71, 1998.

[9] R. Koch, M. Pollefeys, and L. Van Gool. Robust calibration and 3d geometric modeling from large collections of uncalibrated images. In *DAGM*, pages 413–420, 1999.

[10] P. Merrell, A. Akbarzadeh, L. Wang, P. Mordohai, J.-M. Frahm, R. Yang, D. Nistér, and M. Pollefeys. Real-time visibility-based fusion of depth maps. In *Int. Conf. on Computer Vision and Pattern Recognition*, 2007.

[11] P. Narayanan, P. Rander, and T. Kanade. Constructing virtual worlds using dense stereo. In *Int. Conf. on Computer Vision*, pages 3–10, 1998.

[12] T. Sato, M. Kanbara, N. Yokoya, and H. Takemura. Dense 3-d reconstruction of an outdoor scene by hundreds-baseline stereo using a hand-held video camera. *Int. J. of Computer Vision*, 47(1-3):119–129, April 2002.

[13] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. of Computer Vision*, 47(1-3):7–42, April 2002.

[14] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Int. Conf. on Computer Vision and Pattern Recognition*, pages 519–528, 2006.

[15] G. Slabaugh, W. B. Culbertson, T. Malzbender, and R. Schafer. A survey of volumetric scene reconstruction methods from photographs.

[16] G. Turk and M. Levoy. Zippered polygon meshes from range images. In *SIGGRAPH*, pages 311–318, 1994.

[17] M. Wheeler, Y. Sato, and K. Ikeuchi. Consensus surfaces for modeling 3d objects from multiple range images. In *Int. Conf. on Computer Vision*, pages 917–924, 1998.

[18] R. Yang and M. Pollefeys. Multi-resolution real-time stereo on commodity graphics hardware. In *Int. Conf. on Computer Vision and Pattern Recognition*, pages I: 211–217, 2003.