Least Commitment, Viewpoint-based, Multi-view Stereo

Xiaoyan Hu
Stevens Institute of Technology
Hoboken, New Jersey, USA
xhu2@stevens.edu

Philippos Mordohai
Stevens Institute of Technology
Hoboken, New Jersey, USA
mordohai@cs.stevens.edu

Abstract

We address the problem of large-scale 3D reconstruction from calibrated images relying on a viewpoint-based approach. The representation is in the form of a collection of depth maps, which are fused to blend consistent depth estimates and minimize violations of visibility constraints. We adopt a least commitment strategy by allowing multiple candidate depth values per pixel in the fusion process and deferring hard decisions as much as possible. To address the inevitable noise in the depth maps, we explicitly model its sources, namely mismatches and inaccurate 3D coordinate estimation via triangulation, by measuring two types of uncertainty and using the uncertainty estimates to guide the fusion process. To the best of our knowledge, this is the first attempt to model both geometric and correspondence uncertainty in the context of dense 3D reconstruction. We show quantitative results on datasets with ground truth that are competitive with the state of the art.

1. Introduction

Undeniably, there has been significant progress in multi-view 3D reconstruction in terms of accuracy, scalability and more rigorous benchmarking [27, 28]. One question that has not been resolved yet, however, since the answer depends on the specific variant of the problem one is faced with, is which is the “best” approach for multi-view reconstruction. There are methods that achieve outstanding results on single objects surrounded by cameras [13, 3, 26, 37, 20, 38] by taking advantage of silhouettes and the fact that the final surface is a watertight manifold. These assumptions, however, require almost the entire surface of the object to be visible and often relatively easy foreground/background segmentation. In this paper we focus on a different flavor of multi-view stereo where the inputs are images of large-scale scenes that do not provide 360° coverage of the surfaces. In the last few years, several algorithms [12, 23, 30, 9, 11] with moderate processing and storage requirements that do not require any global computations have been instrumental to the success of large-scale, dense 3D modeling [25, 8, 7]. Our approach can also be a component of such systems.

A common characteristic of many problems in computer vision, including 3D reconstruction, is that more regularization is needed when the data are barely sufficient to solve the problem, as for example in binocular stereo which has been addressed via sophisticated global optimization techniques [36, 1]. As more data become available, the need for regularization is reduced since outliers are easier to detect and remove [23, 9] and the solution emerges from the consensus of inliers. To achieve this, it is important that the outliers are random and uncorrelated with the inliers so that systematic errors do not reinforce each other. As long as there is even a

Figure 1. Screenshots of 3D point clouds generated by our method on the data of Strecha et al. [28]. Top: a point cloud from a single depth map of the fountain-P11. Bottom: three overlayed point clouds from Herz-Jesu-P8.
small number of inliers, they cluster around the correct solution, which can be detected because of this consensus. This suggests a least commitment strategy which generates independent hypotheses and infers the final surfaces based on consensus. Regularization in our approach occurs primarily across depth maps and not in pixel neighborhoods. Unlike other methods, we delay making inlier/outlier decisions until we accumulate all evidence for each hypothesis. In fact, we include more than one depth value per pixel in each depth map to account for cases where the true depth was not assigned the maximum photoconsistency score. The difference with other methods that generate multiple candidates per pixel [3, 22, 24] is that they select one of them based on neighborhood and photoconsistency criteria on the same depth map, while our approach allows these candidates to be evaluated on multiple depth maps until the end.

For our method to be applicable to very large image sequences, it should allow the decomposition of the problem into manageable pieces. This can be achieved either by a representation in the form of surface patches in 3D [12, 17, 9] or depth maps [23, 30, 32]. Both types of algorithms, besides depth, also estimate visibility. Patch-based methods often make early hard decisions by assuming that the seeds used to initialize the patches are correct and not considering alternative candidates. Criteria for resolving conflicts between patches include counting the number of images in which a 3D point is visible [9] or selecting the candidate with highest average photoconsistency [12, 17, 9]. Parameters, such as the minimum number of supporting images or the threshold for acceptable photoconsistency, depend on the dataset. We propose a soft way of enforcing long-range free-space and occlusion constraints, similar to [23], that does not require parameter tuning and does not make hard decisions before considering all evidence for and against each depth hypothesis.

In order for our approach to succeed, it must overcome errors due to the small triangulation angle and mismatches. The former cause large uncertainty in the coordinates of the reconstructed 3D points, while the latter introduce spurious depth candidates and may prevent the correct depth value from being among the candidates for a pixel. We address these challenges by explicitly modeling geometric and correspondence uncertainty. Geometric uncertainty is related to the expected error in a depth estimate’s 3D position, given the camera configuration (focal length, resolution, baseline). Here, it is used to determine whether two depth candidates refer to the same part of the surface and, therefore, should be fused. Correspondence uncertainty measures the likelihood of establishing wrong matches on the images. The opposite of correspondence uncertainty will be referred to as confidence throughout the paper. Note that the two types of uncertainty are independent: a depth estimate in the near range has low geometric uncertainty, but may have high correspondence uncertainty due to repetitive texture; conversely, a distinctive point far from the cameras can be matched unambiguously, but it has high geometric uncertainty. To the best of our knowledge, ours is the first dense 3D reconstruction method that considers both types of uncertainty.

Our approach is designed to be part of a large-scale 3D reconstruction system capable of generating high-quality reconstructions in the form of point clouds (Fig. 1) without requiring global processing of the entire sequence which may contain thousands of frames. In this paper, we focus on the core depth map fusion module and not on a complete pipeline encompassing for example pose estimation, view selection and mesh generation. We think that our work provides new insights into a central problem in computer vision and demonstrates that very accurate 3D reconstruction is possible with minimal regularization.

2. Related Work

Seitz et al. [27] categorize multi-view reconstruction algorithms according to whether they represent shape using volumetric grids, global surfaces, depth map collections or surface patches. Due to the requirements for scalability and applicability to large-scale, open surfaces, the latter two categories are of more interest to us.

Methods using depth map collections generate the final 3D surfaces by fusing the input depth maps. This can be achieved by computing signed distance functions on a 3D grid [5, 14, 37] or by generating point clouds [2, 22] and reconstructing meshes from them using the method of [18]. Volumetric fusion does not meet our requirements due to the limitations it imposes on the resolution of the final surface, which is limited by the cubic memory requirement of the grid. Recently, volumetric methods have been scaled up using a two-stage algorithm [35, 16] that estimates the surface by solving a minimum s-t cut problem on the 3D Delaunay mesh of a point cloud extracted by matching keypoints. This allows space away from the surfaces to be represented by very small numbers of tetrahedra.

Alternatively, depth maps can be fused by rendering them on common planes: the images [29, 23, 30, 32] or the ground [11]. Operating on surfaces instead of volumes allows processing at high resolution. The methods of [23] and [32] are of particular interest to us because they take into account the correspondence uncertainty of depth candidates rendered onto a reference view.

Other authors have used individual rays as reference on which to link multiple depth estimates [19, 33] under the assumption that each true depth will be supported by multiple depth maps. Depth linking methods keep track of the uncertainty of the current depth hypothesis and reject unreliable candidates, but do not model long range interactions among points.
Patch-based methods integrate information on small [12, 17, 9] or large patches [21, 34]. Most of them use sparse feature correspondences to initialize patches in 3D, which are filtered to reject outliers and grown according to photoconsistency, which is estimated on several images. Visibility constraints, which are necessary for accurate results, are evaluated by projecting on reference planes, images or patches.

3. Initial Depth and Confidence Estimation

In this section, we describe how the inputs for the fusion process are generated. Depth maps computed from a set of calibrated images using plane-sweeping stereo implemented on the GPU similar to [10] with normalized cross-correlation (NCC) as the similarity function. NCC is computed in windows defined in the reference image and mapped to several target images via a homography through the hypothesized plane. For all experiments in this paper, we sweep four sets of planes: one horizontal and three vertical. One of the vertical sets is fronto-parallel and the other two are rotated 45° clockwise and counter-clockwise around the gravity direction. Horizontal planes are very effective for reconstructing surfaces such as the ground, which are heavily distorted when mapped via homographies defined on vertical planes. (An alternative that approximately models horizontal foreshortening only was proposed by Bradley et al. [2].) Furukawa and Ponce [9], among others, argued for the importance of estimating the normal of the hypothesized planes correctly in achieving high matching accuracy. Our experiments are consistent with their finding, but our approach is faster. We have observed that increasing the sweeping directions has very little impact on the results.

To compute the photoconsistency of a candidate depth \( d_i \) of pixel \((x,y)\) of the reference view for a given plane, all pairwise NCC scores are averaged, excluding cameras where the candidate is invisible (projects out of bounds), and stored in the matching volume. The candidate with the highest NCC among all sweeping directions is selected for each pixel in winner-take-all fashion. When multiple hypotheses per pixel are used in fusion, the local maxima with the highest NCC values are kept regardless of the sweeping direction that generated them.

The second type of inputs to the fusion stage is confidence maps, one for each depth map, that represent the correspondence uncertainty of each depth candidate. Based on our evaluation of confidence measures [15], we selected a form of confidence introduced by Merrell et al. [23], and named AML in [15], to obtain a probability mass function for depth given NCC scores, assuming a uniform depth prior. The confidence of a depth candidate \( d_i \) is defined as follows:

\[
C(d_i) = \frac{e^{-\frac{\text{NCC}(d_0) - \text{NCC}(d_i)}{2\sigma^2}}}{\sum_j e^{-\frac{\text{NCC}(d_0) - \text{NCC}(d_j)}{2\sigma^2}}},
\]

where \( d_0 \) is the depth with the maximum NCC for that sweeping direction. Taking into account \( \text{NCC}(d_0) \) is crucial for allowing confidence values to transfer across depth maps. The choice of (1) was made in part because it allows the assignment of confidence values to depth candidates other than the one with the maximum NCC. Confidence is computed separately per sweeping direction, using the corresponding NCC maximum as \( \text{NCC}(d_0) \). Multiple candidates (different depths) from the same direction compete and decrease each other’s confidence, while if approximately the same depth were selected by multiple directions, this depth receives large support during fusion. By converting photoconsistency to a probability mass function over depth, our algorithm is less sensitive to NCC values which depend on image content.

The fact that we allow the depth probability mass function to be multimodal distinguishes our approach from that of Vogiatzis and Hernández [33] who restrict it to being unimodal. Ambiguity in stereo is manifested as multiple distinct depth candidates which often exist for a pixel and which cannot be modeled by a unimodal distribution. The fusion process of the next section is able to handle a large number of candidate depths per ray of the reference view.

4. Depth Map Fusion

Our depth map fusion algorithm is a synthesis of the confidence and stability-based fusion algorithms [23] and new ideas introduced here. In its design we aimed at eliminating failure modes, heuristics and suboptimal choices of the previous algorithms. We begin by discussing the shortcomings of [23] and how they are addressed here.

Both algorithms of [23], as well as ours, begin by rendering the input depth and confidence maps onto the reference view. (All renderings in this paper are done on points.) They consider violations of visibility constraints, namely occlusions and free space violations, which indicate inconsistencies among the depth maps. An occlusion occurs when a depth map from view \( v \) appears in front of the current depth estimate on the ray of the reference view (Fig. 2(b)), while a free space violation occurs when a depth estimate of the reference view appears in front of depth map \( v \) on the ray of view \( v \) (Fig. 2(c)). Both visibility constraints are required; if only free space violations were considered, the algorithm would select hypotheses far away from the cameras, while the opposite is true for occlusions.

Stability-based fusion [23] seeks a depth that balances the two types of violations. This “median” depth is found by rendering all depth candidates accumulated on the reference view to all other views and evaluating their free space
violations, while occlusions are evaluated on the reference view. The hypothesis that has an equal number of occlusions and free space violations is selected, as it is not too far neither too close to the reference camera according to all the depth maps. The main disadvantage of stability-based fusion is that it only selects one of the existing depth candidates and free space violations is selected, as it is not too far neither too close to the reference camera according to all the depth maps. This is not generally equal to \( D^u(H^v(x, y)) \) which denotes the depth estimated for pixel \((x, y)\) of view \(u\). \( C(H^v(x, y)) \) is the confidence of \( H^v(x, y) \) and \( S(H^v(x, y)) \) is the radius of its support region which is defined below. (These quantities can only be defined with respect to view \(u\) that generated \(H^v(x, y)\).)

**Support.** For each pixel \((x, y)\) of the reference view, we examine the depth hypotheses rendered on it and compute the support each of them receives from the other depth hypotheses for the same pixel (Fig. 2(b)). To do this, we need to address a fundamental computer vision problem: how to distinguish whether two samples are from the same or different surfaces. We accomplish this by examining the geometric uncertainty of a triangulated 3D point. Even though our images are not rectified, a rectifying transformation could have been applied and we can introduce effective disparity, denoted by \(\delta\), which determines the depth of the 3D point given the image coordinates of two corresponding pixels. The relationship between depth and disparity is \(Z = bf/\delta\), where \(f\) is the focal length and \(b\) is the baseline. (We set \(b\) equal to the widest baseline between the reference and any of the target views. Determining the effective baseline for multi-view stereo more precisely is outside the scope of this paper.)

Having written depth as a function of disparity, we can propagate the uncertainty of disparity to determine the uncertainty of depth. Unlike uncertainty in depth which is a function of depth itself, uncertainty in disparity can be considered identically distributed over the entire depth map. Its sources include primarily quantization noise due to limited pixel resolution as well as slight mismatches. Assuming that disparity errors are zero-mean and ignoring higher than second-order moments, the variance of the errors in \(Z\) can be written as a function of \(\sigma_\delta\) as follows [4, 6]:

\[
\sigma_Z^2 = \frac{\partial Z}{\partial \delta} \sigma_\delta^2 \frac{\partial Z}{\partial \delta} = \frac{Z^2}{bf} \sigma_\delta^2 \frac{Z^2}{bf} = \frac{Z^4 \sigma_\delta^2}{b^2 f^2}. \tag{2}
\]

The standard deviation of the error in depth \(\sigma_Z\) grows quadratically with depth. Therefore, the radius within which we will consider two points as being indistinguishable should also grow quadratically with depth.

For each depth hypothesis \(H_i(x, y)\) for pixel \((x, y)\), we seek other depth hypotheses on the same ray that support it by testing whether they are within its support region \(S(H_i)\), which depends on its \(\sigma_{Z_i}\) and a constant \(c_s\). We omit \((x, y)\) for clarity.

\[
C_{\text{supp}}(H_i) = \sum_j C(H_j),
\]

\[
B_i = \sum_j \frac{C(H_i)H_j}{\sum_j C(H_j)},
\]

if \(|D^{ref}(H_i) - D^{ref}(H_j)| \leq S(H_i) = c_s\sigma_{Z_i},\)
Blended hypotheses are penalized if they are also recorded. We subtracted from the updated confidence of Blended hypotheses, the confidence of the Blended hypothesis is and thus that they may potentially be wrong. For each occlusion indication that they may not be visible from that viewpoint, occluded on the ray of the reference view since this is an indication that they may not be visible from that viewpoint, and thus that they may potentially be wrong. For each occlusion, the confidence of the occluding depth estimate is subtracted from the updated confidence of \( B_i \).

\[
C_{occ}(B_i) = C_{supp}(B_i) - \sum_j C(H_j), \tag{4}
\]

where \( S(B_i) \) is taken as referring to \( S(H_i) \). Hypotheses that support \( B_i \) are not considered to be occluding it.

**Free space violations.** Blended hypotheses are then rendered onto all depth maps to assess whether they violate the free space of the estimated surfaces. Let us denote the projection of a 3D point onto view \( v \) as \( P^v(B_i) = (x^v, y^v) \) and its depth with respect to view \( v \) as \( D^v(P^v(B_i)) \). If a blended hypothesis \( B_i \) is in front of the current depth estimate \( D_i^v(x^v, y^v) \) by more than its support range, then a violation occurs and the blended hypothesis is penalized by the confidence value at \( (x^v, y^v) \) in the confidence map for view \( v \). This test is repeated for all views.

\[
C_{fin}(B_i) = C_{occ}(B_i) - \sum_v C^v(x^v_i, y^v_i), \tag{5}
\]

where \( S(B_i) \) is negative, then it is also rejected and no depth is assigned to the pixel. Figure 3 shows an example of fusion. The resulting depth map is iteratively filtered by a median filter to fill in the holes. At least 50% of the pixels in the window must have valid depths for a hole at the center to be filled. Valid depths are not altered.

A final, but important, note in this section is that the algorithm does not change when multiple candidates for each pixel are used. In fact, we run the code without modification by providing as input three “different” depth, confidence and geometric uncertainty maps for the same camera. Allowing these additional hypotheses to compete and support other hypotheses is precisely what our objective was.

5. Experimental Results

In this section, we present an evaluation of our algorithm on two real outdoor datasets with ground truth\(^1\) [28]. We rendered the provided 3D models to generate ground truth depth maps for fountain-P11 and Herz-Jesu-P8, which contain 11 and 8 images respectively. All images were downsampled to \( 1536 \times 1024 \) and experiments were performed with fixed parameters: three adjacent images were used for the computation of each depth map, the NCC window was \( 7 \times 7 \), \( \sigma \) in (1) was set to 0.2 and the sweeping planes where distributed in space so that the disparity step between them was no more than 0.2, with disparity defined between the reference view and the farthest target view. The derivative of depth with respect to disparity (2) was used to determine the steps between planes. All depth maps are fused on each view, except for the two extreme views of each dataset. The support range was set to \( 4 \sigma Z_s \), i.e. \( c_s = 4 \) in (4), and median filtering for hole filling was performed in \( 13 \times 13 \) windows.

Due to the high resolution of the input images, storing the entire cost volume on the GPU is impossible. While depth can be computed by just keeping track of the current best candidate, computing confidence according to (1) requires the entire volume which has to be computed in parts and transferred to CPU memory. To avoid this very time consuming operation, we approximate confidence computation by explicitly maintaining the three highest NCC scores for every pixel and histogramming the remaining NCC values in histograms of 16 bins. This allows us to only transfer the top depth candidates and their associated confidence values from the GPU to the CPU resulting in a major speedup with negligible loss of accuracy.

We also compare our final fused depth maps with results kindly provided to us by their authors. 3D models were given to us by Furukawa and Ponce [9], Zaharescu et al. [38], Tylecek and Sara [31] and Jancosek and Pajdla [16] and they are denoted by FUR, ZAH, TYL and JAN, respec-

---

\(^1\)The online evaluation system for these data was unavailable for several months before the submission deadline.
tively, in the remainder. We label our results with LC, for “least commitment”. We are grateful to all the authors for making their models available to us.

We use two types of errors to evaluate the depth maps: absolute errors which are defined as the absolute values of the difference between depth estimates and the corresponding ground truth depths and relative errors which are defined as:

\[
    e_{rel} = \frac{|Z - Z_{GT}|}{Z_{GT}},
\]

where \(Z\) is the estimated depth and \(Z_{GT}\) is the ground truth, \(f\) is the focal length of the reference camera and \(b\) is the widest baseline between the reference and either of the target views, as in (2). Relative errors show how well the surface is reconstructed taking into account camera configuration and distance to the scene. They can be viewed as measures of the effective standard deviation of noise in pixel correspondences that would give rise to the observed 3D errors. For example, \(e_{rel} = 4\) means that the effective matching error, due to quantization, calibration errors and failures of the matching function, was 4 pixels. We use cumulative histograms, similar to those of [28], of both types of errors to compare the results of all methods. Note, however, that the standard deviation of the LIDAR data used in [28] was not available to us.

Table 1 presents quantitative results comparing various stages of our algorithm, as well as our results with those of other algorithms. Only pixels with ground truth depth are considered. The table contains statistics of absolute and relative errors for: i) raw depth maps (top candidate per pixel according to confidence), ii) fused depth maps using one depth candidate per pixel, iii) fused depth maps using three candidates per pixel and iv) median-filtered fused depth maps using three candidates per pixel (LC), which is our final result.

The error statistics of fused depth maps in which one or three candidates per pixel were used are similar, but always in favor of using more candidates, the qualitative difference is significant. This is because using multiple candidates is more effective near depth discontinuities where stereo is prone to errors. See Fig. 4 for examples from the fountain-P11 dataset.

To compare our results with those of other methods, meshes were rendered onto the image planes to generate depth maps and all reported errors are computed on the pixels of these depth maps. Figure 5 shows cumulative histograms of relative error for all methods compared to the ground truth. (Cumulative histograms of absolute error are similar and have been omitted due to space constraints.) Table 1 contains statistics for a few thresholds on absolute and relative errors. The most salient conclusion is that the method of Jancosek and Pajdla [16] ranks first. This can be observed in the results published in [35, 16]. Our method ranks clearly second on fountain-P11, while the other methods form essentially one cluster. On Herz-Jesu-P8, our method forms a cluster with TYL and FUR, outperforming the latter slightly for all thresholds and being better than the former for larger thresholds. It should be noted that the ground truth provided with Herz-Jesu-P8 does not include the handrails next to the steps and the horizontal bar above the left entrance (see Fig. 6), which our method is able to reconstruct. TYL reconstructs smaller pieces of these parts, but their overall effect on error statistics is hard to assess. Our method produces depths for more pixels and thus achieves higher density at large values of the threshold. Results by other methods can be seen at the currently inactive

### Table 1

<table>
<thead>
<tr>
<th></th>
<th>Error</th>
<th>2cm</th>
<th>10cm</th>
<th>(\sigma)</th>
<th>3(\sigma)</th>
</tr>
</thead>
<tbody>
<tr>
<td>fountain-P11</td>
<td>Raw</td>
<td>0.695</td>
<td>0.784</td>
<td>0.732</td>
<td>0.838</td>
</tr>
<tr>
<td></td>
<td>Fuse-1</td>
<td>0.740</td>
<td>0.915</td>
<td>0.760</td>
<td>0.874</td>
</tr>
<tr>
<td></td>
<td>Fuse-3</td>
<td>0.749</td>
<td>0.919</td>
<td>0.768</td>
<td>0.876</td>
</tr>
<tr>
<td></td>
<td>LC</td>
<td>0.754</td>
<td>0.930</td>
<td>0.774</td>
<td>0.886</td>
</tr>
<tr>
<td></td>
<td>FUR</td>
<td>0.731</td>
<td>0.838</td>
<td>0.760</td>
<td>0.828</td>
</tr>
<tr>
<td></td>
<td>ZAH</td>
<td>0.712</td>
<td>0.832</td>
<td>0.732</td>
<td>0.818</td>
</tr>
<tr>
<td></td>
<td>TYL</td>
<td>0.732</td>
<td>0.822</td>
<td>0.754</td>
<td>0.811</td>
</tr>
<tr>
<td></td>
<td>JAN</td>
<td>0.824</td>
<td>0.973</td>
<td>0.842</td>
<td>0.948</td>
</tr>
<tr>
<td>Herz-Jesu-P8</td>
<td>Raw</td>
<td>0.584</td>
<td>0.781</td>
<td>0.695</td>
<td>0.776</td>
</tr>
<tr>
<td></td>
<td>Fuse-1</td>
<td>0.638</td>
<td>0.817</td>
<td>0.735</td>
<td>0.811</td>
</tr>
<tr>
<td></td>
<td>Fuse-3</td>
<td>0.637</td>
<td>0.822</td>
<td>0.739</td>
<td>0.817</td>
</tr>
<tr>
<td></td>
<td>LC</td>
<td>0.649</td>
<td>0.848</td>
<td>0.757</td>
<td>0.841</td>
</tr>
<tr>
<td></td>
<td>FUR</td>
<td>0.646</td>
<td>0.836</td>
<td>0.746</td>
<td>0.837</td>
</tr>
<tr>
<td></td>
<td>ZAH</td>
<td>0.220</td>
<td>0.501</td>
<td>0.377</td>
<td>0.533</td>
</tr>
<tr>
<td></td>
<td>TYL</td>
<td>0.658</td>
<td>0.852</td>
<td>0.788</td>
<td>0.853</td>
</tr>
<tr>
<td></td>
<td>JAN</td>
<td>0.739</td>
<td>0.923</td>
<td>0.831</td>
<td>0.912</td>
</tr>
</tbody>
</table>

Figure 4. Details of fused depth maps for fountain-P11 before filtering, using one (left) and three (right) depth candidates per pixel. Boundaries are reconstructed more precisely using more candidates.
web site of [28]².

Revisiting one of the questions we set out to answer in this paper, our method seems to outperform FUR on both datasets, while it produces denser depth maps by reconstructing areas of high uncertainty that are bypassed by FUR. We speculate that this is due to not initializing patches because reliable seams could not be detected.

For visualization purposes, we generated colored point clouds using the depth maps and captured screen shots shown in Figs. 1 and 6.

All experiments were performed on a PC with a quad-core Intel i7-920 at 2.67GHz, 6GB of RAM and an Nvidia GTX570 with 1.3GB of memory. Computing a $1536 \times 1024$ depth map for fountain-P11 using on average 8309 planes and NCC in $7 \times 7$ windows takes 320 sec on the GPU using OpenCL. The same operation for Herz-Jesu-P8 requires an average of 6717 planes and takes 268 sec. Fusion of 27 depth maps of fountain-P11 (3 depth maps for each of the 9 views) takes 182 sec, while it takes 73 sec for 18 depth maps for Herz-Jesu-P8 using C++ on the CPU. Median filtering in $13 \times 13$ windows takes 22.6 sec in Matlab. Porting all steps to the GPU, further optimizing the code and investigating the effectiveness of subpixel approximations are among our priorities for future work.

6. Conclusions

We have presented a method for multi-view 3D reconstruction that, compared to other methods, delays the transition to a world-based representation and most importantly delays hard decisions on the correctness of depth hypotheses. We have shown that viewpoint-based approaches can effectively aggregate information and take into account long range interactions, in the form of occlusions and free-space violations, which are harder to implement using 3D representations. Experiments on data with ground truth show that viewpoint-based methods, which have fallen out of favor recently, can compete with the state of the art in multi-view reconstruction while being faster.

Our analysis of relative errors can be used to characterize the expected performance of multi-view stereo systems as a function of depth given the configuration parameters (focal lengths and baselines). If the results of ZAH on Herz-Jesu-P8 are excluded, all methods perform consistently in terms of relative error. The same cannot be said for absolute error since it strongly depends on the actual depth. Extending this preliminary study to more diverse datasets will hopefully lead to empirical sensor models, i.e. expected error distributions for a given depth, for stereovision sensors.

Acknowledgements

We are grateful to the authors of [9, 38, 31, 16] for providing their results to us. This research has been supported in part by the Domestic Nuclear Detection Office of the U.S. Department of Homeland Security, by the National Science

---

²http://cvlab.epfl.ch/~strecha/multiview/denseMVS.html is still online
References


