Local Readjustment for High-Resolution 3D Reconstruction

Siyu Zhu[1], Tian Fang[2], Jianxiong Xiao[3], and Long Quan[4]
[1,2,4]The Hong Kong University of Science and Technology
[3]Princeton University

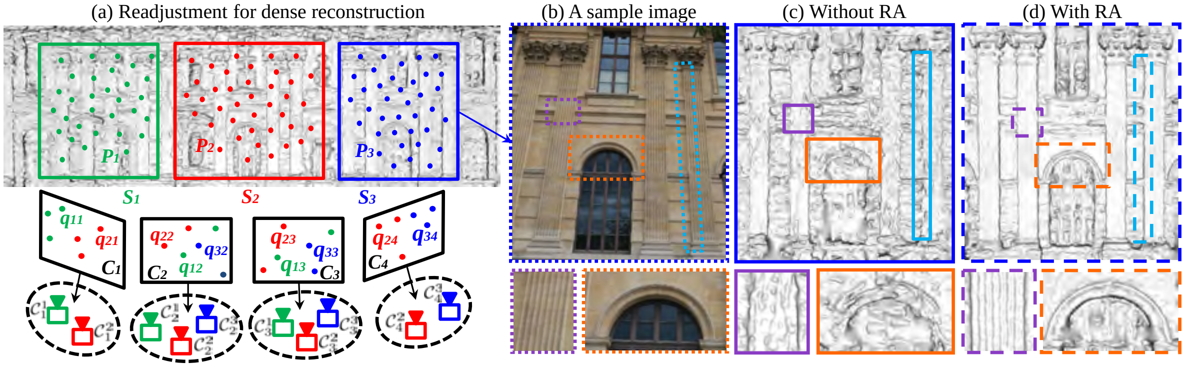


Figure 1. Taking SfM points and camera poses as inputs, we first decompose the points into segments, and then re-optimize each individual segment and its corresponding local cameras. This significantly improves the reconstruction quality for fine geometry details.

Abstract

Global bundle adjustment usually converges to a non-zero residual and produces sub-optimal camera poses for local areas, which leads to loss of details for high-resolution reconstruction. Instead of trying harder to optimize everything globally, we argue that we should live with the non-zero residual and adapt the camera poses to local areas. To this end, we propose a segment-based approach to readjust the camera poses locally and improve the reconstruction for fine geometry details. The key idea is to partition the globally optimized structure from motion points into well-conditioned segments for re-optimization, reconstruct their geometry individually, and fuse everything back into a consistent global model. This significantly reduces severe propagated errors and estimation biases caused by the initial global adjustment. The results on several datasets demonstrate that this approach can significantly improve the reconstruction accuracy, while maintaining the consistency of the 3D structure between segments.

1. Introduction

Typically multi-view 3D reconstruction pipeline requires Structure from Motion (SfM) to solve for the camera poses using bundle adjustment [26], in which all the 3D structures and cameras are simultaneously optimized to obtain a Maximum Likelihood Estimation. This solution is theoretically optimal [26] in terms of minimal variance. However, in practice, the residual error is never minimized to zero. Therefore, such a global adjustment does not guarantee an optimal estimation for local areas, which is critical for high-resolution 3D reconstruction.

In particular, the input images are usually taken under different conditions of lighting, scale, surface reflection, and weather, using various cameras and lens with different focus, sensor noise and distortion, which may not be modeled perfectly during bundle adjustment. Perturbations among images are therefore no longer uniform. The over-simplified assumption on globally uniform perturbations can lead to loss of reconstruction details. Second, because of the inter-connectivity between all the camera and point parameters in bundle adjustment, the global estimation will distribute a local error over all estimated parameters, which biases the estimation of other parameters locally. In the extreme case, the mismatched point features in a local region may contaminate the detailed geometry of other correctly matched region. Furthermore, uneven viewpoint distributions and non-uniformly spaced 3D points and point correspondences also result in local and biased estimations.

Instead of trying hard to globally adjust space structures and camera poses all together, in this paper, we embrace the fact that non-uniformed perturbations can never be optimized perfectly on a global scale. We propose a segment-based bundle adjustment approach that divides sparse SfM

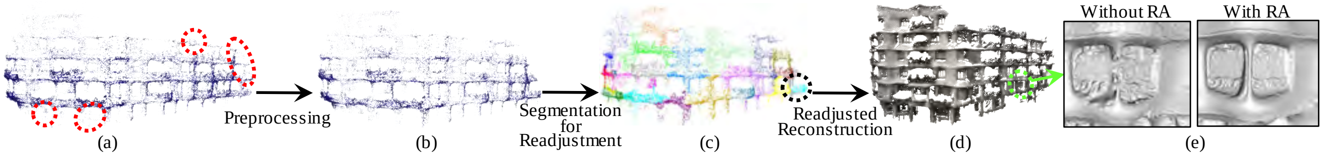


Figure 2. The pipeline of our approach. (a) shows the initial SfM points. (b) shows the SfM points after preprocessing. We note that 3D points marked by red dashed circles in (a) are filtered. (c) shows the result of 3D point segmentation. Note that the segment marked by a black dashed circle is an invalid segment because of insufficient 3D points. (d) shows the dense reconstruction result using the method described in [19]. (e) shows the comparison between mesh models with and without readjustment (RA).

points into smaller and well-conditioned segments for re-optimization, which are still able to be fused back into a consistent global model. The key to this approach is the introduction of *local cameras*, which are inherited from globally optimized cameras (*global cameras*) and locally re-optimized in their corresponding segments (see Figure 1). Since local cameras are re-optimized in well-conditioned segments with evenly spaced 3D points and uniform perturbations, and we only choose the segments of high accuracy for re-optimization, estimation biases and severe errors propagated from other portions of the reconstruction can be eliminated, and detailed geometry is recovered.

The contributions of our approach are three-fold. First, we introduce a local readjustment mechanism that prevents any errors of poorly conditioned regions from being propagated to other regions. Second, we propose a carefully designed segmentation algorithm to make the local segments well-conditioned for re-optimization. Finally, we propose a boundary fusion method that ensures the re-optimization gives consistent and remarkably improved results.

1.1. Related Work

Many great works have been done to improve the accuracy of geometry in multi-view 3D reconstruction. The first type of approach is to optimize camera calibrations, which is related to our work. By adjusting the 3D position of the markers, the inaccuracy of a calibration chart can be compensated [18]. In [16] and [29], the silhouette information is utilized instead. Furukawa *et al.* [9] present a novel method for accurate camera calibrations using top-down information from rough camera parameter estimations and the output of Patch-based Multi-view Stereo Software [10] to search for additional image correspondences. Similarly, Goldlücke *et al.* [14] and Aubry *et al.* [2] refer to super-resolution textures or meshes for variational camera geometry refinement. In [27], estimates of depth and visibility in a series of selected cameras are optimized iteratively. However, these methods are not ideally suited to general or large-scale scenes due to their dependency on scene information or the outputs of multi-view stereo systems.

Some depth-map based methods control the accuracy of geometry in multi-view stereo by appropriately merging and cleaning up the depth maps. Goesele *et al.* propose a method [13] that selects images with the most compatible resolution to handle variations in the sampling resolutions of images. Bailer *et al.* [3] extend this approach to handle huge unstructured datasets. Gallup *et al.* [11] use an image pyramid to select an appropriate baseline and resolution to preserve depth accuracy. Campbell *et al.* [4] use a discrete label MRF optimization to remove the outliers from the depth-maps and improve the quality of depth-maps used for multi-view stereo, while such approaches commonly handle the accuracy variation by selecting appropriate images prior to reconstruction. We instead introduce local cameras for local fitting of different groups of SfM points to handle the variation in point accuracy.

Another type of approach is to directly optimize the dense reconstructed points or meshes. Furukawa *et al.* [8] use quality and visibility filters to filter out low resolution geometry and merge multi-view stereo reconstruction. In [10], reconstructed meshes are refined by both photometric consistency and regularization constraints. The authors of [28] capture small details by a mesh-based variational refinement. These techniques control the accuracy of the geometry posterior to dense reconstruction and may be used in conjunction with our proposed method.

As the key idea of our approach, divide-and-conquer methods are commonly used to handle scalability problems in bundle adjustment. Steedly *et al.* [24] introduce a spectral partitioning approach, which divides the entire large bundle problem of sequential images into small pieces for easier operation while preserving the low error modes of the original system. Ni *et al.* [21] propose an out-of-core bundle adjustment algorithm, which decouples the original problems into sub-maps with their own local coordinate systems. This work is similar to ours but it has not directly addressed the problem of initialization for the sub-maps. In [22], subproblems within a relative coordinate system are tackled rather than a problem in a consistent global coordinate system. Instead of at the image level operated by most other works, the authors of [7] and [12] recover 3D structure hierarchically and apply divide-and-conquer at the variable level. In summary, these algorithms generally aim at making large-scale bundle problems manageable and have not directly addressed the problem of severe error propagation and estimation biases.

2. Overview

The readjustment approach consists of several stages. First, we use SfM points and their corresponding cameras as well-conditioned initialization and filter out the 3D points with low accuracy or abnormal distributions (Section 2.1). Next, 3D points are divided into well-conditioned segments (Section 3) and those with high accuracy and enough 3D points can be further re-optimized using segment-based bundle adjustment (Section 4). Finally, 3D points and their corresponding local cameras can be used for dense reconstruction in each segment separately and a boundary fusion method is used to merge 3D points reconstructed from different camera clusters into a consistent model (Section 4).

2.1. Preprocessing

Let $P = \{P_i\}$ be a set of 3D points and $C = \{C_j\}$ their corresponding cameras. Assume that $\{P_i\}$ and $\{C_j\}$ have been globally optimized by existing bundle adjustment algorithms (called *global bundle adjustment*). Global bundle adjustment is crucial for our approach in two ways. First, it provides well-conditioned initial values for segment-based bundle adjustment, since local cameras regard the globally optimized camera parameters as their initial values. Without well-conditioned initialization, segment-based bundle adjustment may lead to great inconsistencies in 3D structure and local cameras between adjacent segments. Second, for invalid segments, which are either too small or of low accuracy, globally optimized 3D structure and camera parameters are regarded as their local parameters.

When a bundle problem is divided into smaller subproblems, propagated errors may cause more severe perturbations in local re-optimization systems than in global systems. Therefore, we should filter out the 3D points of extremely low accuracy. To quantitatively evaluate 3D point accuracy, we use the uncertainty covariance of the 3D point position. Here, we choose the normal covariance introduced in [20] and a fast computation method [19] to compute the covariance matrix. We should note that all mentions of 3D point accuracy $u(P_i)$ for a given 3D point P_i are expressed as the trace of its covariance matrix, $tr(Cov_{P_i})$, in this paper. Therefore, we regard these: $u(P') > \alpha E(u(P_i))$ ($\alpha = 2.2$ in all our experiments), as 3D points of low accuracy and tend to filter them out. In the following sections, $E(X)$ is the mathematical expectation for $x_i \in X$ and $E_k(X)$ is the mathematical expectation for $x_i \in X_k$ and $X_k \subset X$. Furthermore, we also remove some isolated 3D points for more reliable 3D point segmentation.

3. Segmentation for Readjustment

3.1. Segmentation Constraints

The motivation of our 3D point segmentation algorithm is to divide SfM points into well-conditioned segments for re-optimization. Based on our observations, ”well-conditioned” segments $\{S_k\}$ generally satisfy the following three constraints: (1) 3D points in each segment have similar accuracy (*accuracy constraint*); (2) 3D points are uniformly and continuously distributed in a segment (*density constraint*); and (3) the number of 3D points in one segment is enough to fully constrain local cameras (*size constraint*).

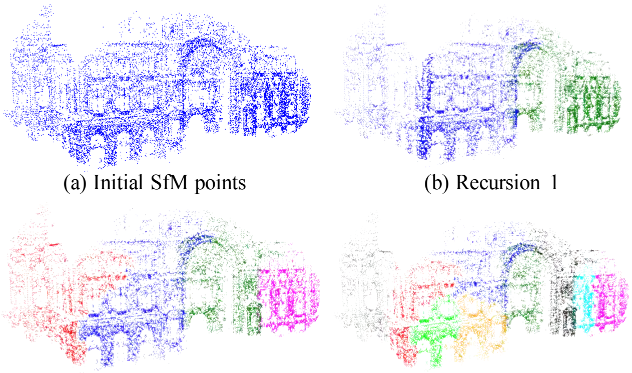

(a) Initial SfM points
(b) Recursion 1
(c) Recursion 2
(d) Recursion 4

Figure 3. Demonstration of 3D point segmentation algorithm. (a) shows the initial SfM points after preprocessing. (b) and (c) show the results after the first and second recursion of segmentation, respectively. (d) shows the final segmentation result.

Accuracy Constraint Intuitively, groups of 3D points with large perturbations tend to perturb ones with small perturbations in bundle problems. Therefore, we should constrain the upper-bound of variance of point parameter uncertainties within a segment: $\forall k, E_k(\|u(P_i) - \overline{u_k}\|^2) \leq \lambda_u E(\|u(P_i) - \overline{u}\|^2)$, where $u(P_i) = tr(Cov_{P_i})$, $\overline{u_k} = E_k(u(P_i))$ for $P_i \in S_k$, and $\overline{u} = E(u(P_i))$.

Density Constraint To avoid local and biased estimations, 3D points in the same segment should have uniform distributions. Therefore, in the same segment, the variance of point density should satisfy: $\forall k, E_k(\|d(P_i) - \overline{d_k}\|^2) \leq \lambda_d E(\|d(P_i) - \overline{d}\|^2)$, where $d(P_i) = 1/V(k)$ is the local density around P_i, $V(k)$ is the volume of the bounding ball of its k-nearest points ($k = 10$), $\overline{d_k} = E_k(d(P_i))$ for $P_i \in S_k$, and $\overline{d} = E(d(P_i))$.

Size Constraint On the one hand, small sized segments generally have more uniform point distributions and perturbations. While on the other hand, small sized segments are vulnerable to severe errors and outliers (though very few after preprocessing) and sometimes cannot fully constrain local cameras. To avoid potential degeneracies in segment-based bundle adjustment, the number of 3D points in each segment, $|S_k|$, must satisfy the following constraint: $\forall k, |S_k| > \lambda_s$. Moreover, to guarantee the reliability of segment-based bundle adjustment, we hope the number of 3D points in a segment is as large as possible while satisfying the accuracy and density constraints.

In summary, we regard the following three constraints as the termination conditions of our segmentation algorithm:

$$\forall k, E_k(\|u(P_i) - \overline{u_k}\|^2) \leq \lambda_u E(\|u(P_i) - \overline{u}\|^2),$$

$$\forall k, E_k(\|d(P_i) - \overline{d_k}\|^2) \leq \lambda_d E(\|d(P_i) - \overline{d}\|^2),$$

$$\forall k, |S_k| > \lambda_s,$$

where $\lambda_u = 0.45$, $\lambda_d = 0.8$, and $\lambda_s = 1000$ in this paper. These three parameters determine the number of segments and empirically generate satisfactory segmentation results. As stated in the experimental section, since the acceptable number of segments has a wide range, the final results are not that sensitive to these parameters.

3.2. Segmentation Process

We regard this segmentation as a weighted graph labeling problem and use joint affinity measures to construct the 3D graph. As an approximate solution, we recursively use the Normalized Cuts algorithm until the accuracy and density constraints are satisfied or the size constraint is violated. Indeed, our approach is not globally optimal, but we observe in our experiments that our method works satisfactorily (See Figure 4 for the 3D point segmentation algorithm).

Graph Construction The set of edges, which define the affinity, are constructed using the k-Nearest Neighbor (k-NN) technique according to the 3D Euclidean distance, and k is set to 5 by default. The edge weight between two 3D points P_i and P_j is defined as

$$a(P_i, P_j) = a_u(P_i, P_j) \cdot a_d(P_i, P_j) \cdot a_e(P_i, P_j),$$

where $a_u(P_i, P_j)$ is the accuracy affinity, $a_d(P_i, P_j)$ the density affinity, and $a_e(P_i, P_j)$ the Euclidean distance affinity.

First, 3D points with similar accuracy tend to belong to the same group and we use the accuracy affinity encoding the difference in accuracy as $a_u(P_i, P_j) = exp(-\frac{\|u(P_i)-u(P_j)\|^2}{2{\sigma_u}^2})$, where $u(P_i) = tr(Cov_{P_i})$ and $\sigma_u^2 = E(\|u(P_i) - u(P_j)\|^2)$. In addition to point accuracy, the uniform distribution of points is also crucial for the parameter re-optimization of 3D points. We incorporate the difference between point densities into the affinity and define $a_d(P_i, P_j) = exp(-\frac{\|d(P_i)-d(P_j)\|^2}{2{\sigma_d}^2})$, where $d(P_i)$ is the point density of P_i, and $\sigma_d^2 = E(\|d(P_i) - d(P_j)\|^2)$. To guarantee continuous point distribution, closer points in space generally have a higher probability of belonging to the same group. We naturally take the 3D Euclidean distance as an affinity measure $a_e(P_i, P_j) = exp(-\frac{\|e(P_i)-e(P_j)\|^2}{2{\sigma_e}^2})$, where $e(P_i)$ is the coordinate of 3D point P_i, and $\sigma_e^2 = E(\|e(P_i) - e(P_j)\|^2)$.

```
Input: SfM points $\{P_i\}$ after preprocessing, accuracy threshold $\lambda_u$,
density threshold $\lambda_d$, and size threshold $\lambda_s$.
Initialize 3D point segments: $S_0 \leftarrow \{P_i\}$ and $S_0$ is unmarked;
For each unmarked 3D point segment $S_k$
  If $S_k$ satisfies accuracy, density and size constraints ($\lambda_u$, $\lambda_d$, $\lambda_s$)
    If $S_k$ passes the segmentation evaluation ($\theta$)
      Mark $S_k$ as valid;
    Else
      Mark $S_k$ as invalid;
  Else if $S_k$ dose not satisfy size constraint ($\lambda_s$)
    Mark $S_k$ as invalid;
  Else
    Divide $S_k$ into two unmarked segments: $S_{k_1}$ and $S_{k_2}$;
Output: 3D point segments $\{S_k\}$.
```

Figure 4. 3D point segmentation algorithm.

Segment Division Next, we recursively use Normalized Cuts to partition the 3D points to satisfy the accuracy and density constraints. In the meantime, if a segmented part violates the size constraint, the subdivision of this segment stops. In summary, the division of a segment repeats until the accuracy and density constraints are satisfied or the size constraint is violated. We note that our segmentation approach tends to easily and rapidly satisfy the constraints while achieving acceptable segmentation results in most cases. Although an iterative algorithm can be introduced for optimization, it is computationally expensive for large scale 3D points and its convergence cannot be theoretically guaranteed.

Finally, one important issue is that adjacent segments should have small overlap to guarantee that the finally reconstructed 3D points in each isolated segment can be well fused together without obvious discontinuity. Technically speaking, we expand each segment by taking over several 3D points of its adjacent segments (10% overlap works satisfactorily in our implementation).

Segmentation Evaluation Now, we should evaluate the final segments and find those of high accuracy and satisfying the size constraint (valid segments) for re-optimization in segment-based bundle adjustment. First, the segments violating the size constraint in the previous step are obviously invalid. As for the segments satisfying the size constraint, re-optimizing these with low accuracy does not remarkably improve the final results, and they are also regarded as invalid. Mathematically, segments of low accuracy generally have large average uncertainty covariance of point positions and we can conclude: the segment S_k is invalid if $\overline{u_k} > \theta\overline{u}$, where $\overline{u_k} = E_k(u(P_i))$ for $P_i \in S_k$, $\overline{u} = E(u(P_i))$, $u(P_i) = tr(Cov_{P_i})$, and $\theta = 1.2$ in our implementation.

Finally, the invalid segments do not perform further optimization and we regard the globally optimized camera parameters as their local parameters. In each of the valid segments, segment-based bundle adjustment is performed to re-optimize the 3D structure and local camera parameters separately.

Dataset		Real datasets without ground truth									Real datasets with ground truth						Synthetic datasets		
		Station	Casa Milla	Street	Louvre			Castle			f-P11	H-P8	e-P10	c-P19	H-P25	c-P30	Depot	Block	Tower
# of images		39	61	68	26	53	101	120			11	8	10	19	25	30	40	80	120
Image resolution		21M	21M	21M	21M			5.3M	10.5M	21M	6.3M	6.3M	6.3M	6.3M	6.3M	6.3M	21M	21M	21M
# of SfM points	Input	18662	31730	57699	16931	35163	103291	49210	51620	81267	13064	7703	9746	15236	20991	23651	29671	67421	149874
	After filtering	17945	28590	51119	16003	33537	97769	48931	51499	78174	12954	7678	9645	14987	20612	22763	28315	64877	147531
# of segments		9	22	26	10	19	34	30	29	46	6	3	5	8	14	16	21	37	52
# of valid segments		8	20	25	9	18	32	27	27	43	6	3	5	8	13	16	20	35	48
Running time [secs]	Global BA	51.2	97.4	160.6	41.1	181.7	678.7	359.6	232.2	248.0	32.4	21.7	29.7	42.6	59.0	88.1	126.4	198.2	471.0
	Segmentation	5.0	10.6	16.6	3.7	9.5	49.5	67.6	82.3	89.8	3.2	1.7	2.3	4.6	6.4	9.3	9.7	18.4	52.2
	Segment BA	7.8	6.9	11.2	3.4	6.3	18.0	9.9	12.5	15.2	3.9	3.7	2.0	6.6	8.3	8.2	5.5	13.4	26.1

f-P11: fountain-P11 H-P8: Herz-Jesu-P25 e-P10: entry-P10 c-P19: castle-P19 H-P25: Herz-Jesu-P25 c-P30: castle-P30

Table 1. General statistics of the datasets and algorithms. Notations: The postfixes of the Louvre_26, Louvre_53, and Louvre_101 datasets correspond to the number of images in the datasets and those of the Castle_5M, Castle_10M, and Castle_21M datasets correspond to the resolution of images in the datasets. The epipolar error is the average distance between the data points and corresponding epipolar lines.

4. Readjusted Reconstruction

Segment-Based Bundle Adjustment Now, each point segment S_k corresponds to a set of local cameras $\{C_j^k\}$, where for $\forall \mathcal{C}_j^k$ related to certain S_k, $\exists \mathcal{P} \subset S_k$ satisfying that $\mathcal{P}$ are visible in $\mathcal{C}_j^k$. In valid segments, we first use global camera parameters to initialize local camera parameters, and then bundle adjustment is performed for each valid segment. Since 3D points and their corresponding local cameras in each segment are optimized separately, they achieve the optimum in corresponding well-conditioned segments and compensate remarkably for any estimation biases and severe propagated errors of global bundle adjustment.

Dense Reconstruction Next, we regard the 3D structure and the corresponding local cameras of each segment as an autonomous unit for dense reconstruction, which can obviously be processed in parallel. Intuitively, most existing multi-view stereo algorithms can be used to reconstruct dense 3D points, including the greedy expansion approach [10, 19], and the variational multi-view stereovision [28].

Boundary Fusion Since dense 3D points are propagated in each segment independently, the overlapping regions between adjacent segments are reconstructed by slightly inconsistent local cameras from different segments. Inspired by [6, 8], we propose to use an efficient filter to fuse reconstructed points and improve the reconstruction quality of the overlapping region. More concretely, to guarantee a uniform distribution of high quality 3D points in the overlapping region, our filter tends to preserve those of high accuracy and low density. Quantitatively, we give each 3D point P_i a weight $w(P_i)$ as:

$$w(P_i) = w_a(P_i) \cdot w_c(P_i),$$

where $w_a(P_i)$ is the accuracy term and $w_c(P_i)$ is the completeness term. Here, we use the function $f(P_i, C_i)$ introduced in [8], which takes baselines and pixel sampling rates into account to measure the accuracy of 3D point P_i achieved by its visible images C_i, as $w_a(P_i)$, and $w_c(P_i)$ is the inverse of the local density around P_i. In our implementation, we repeatedly discard the 3D point with the lowest weight until the density of the overlapping region equals the average density of its corresponding segments.

5. Experimental Results

5.1. Implementation

Our approach has been implemented in C++ on a PC with Quad-Core Intel 3.10GHz processor for all our experiments. All the datasets have been calibrated by the standard automated SfM approach as described in [15]. The input of our approach can either be sparse 3D points produced by a SfM system (e.g. Bundler [23]) or sparse samples of the quasi-dense reconstruction (e.g. [6, 19]). Both the global and segment-based bundle adjustment are handled by Ceres Solver [1], and Normalized Cuts by Graclus [5].

5.2. Datasets

Table 1 provides some general statistics of the datasets and algorithms. Note that, to fully demonstrate the satisfactory performance of our approach, we have tested our method on three types of datasets: real datasets with ground truth, real datasets without ground truth, and synthetic datasets.

Obviously, the real datasets with ground truth use ground truth data to quantify the absolute 3D accuracy and demonstrate any improvements in camera geometry. Here, we use the well-known dense multi-view stereo benchmark [25].

Since current real datasets with ground truth are generally limited to small-scale reconstruction scenes, we also use some real datasets without ground truth but greater numbers of images of higher resolution to qualitatively validate the improvement in 3D accuracy. To explore the effects of the dataset scales, measured by the number of images and image resolution, on the improvement in 3D accuracy, we randomly select two subsets of the sequential images from the Louvre dataset and make up three datasets: the Louvre_26, Louvre_53 and Louvre_101 datasets, and downsample the pixels of the original images in the Castle dataset resulting in three datasets: the Castle_5M, Castle_10M, and

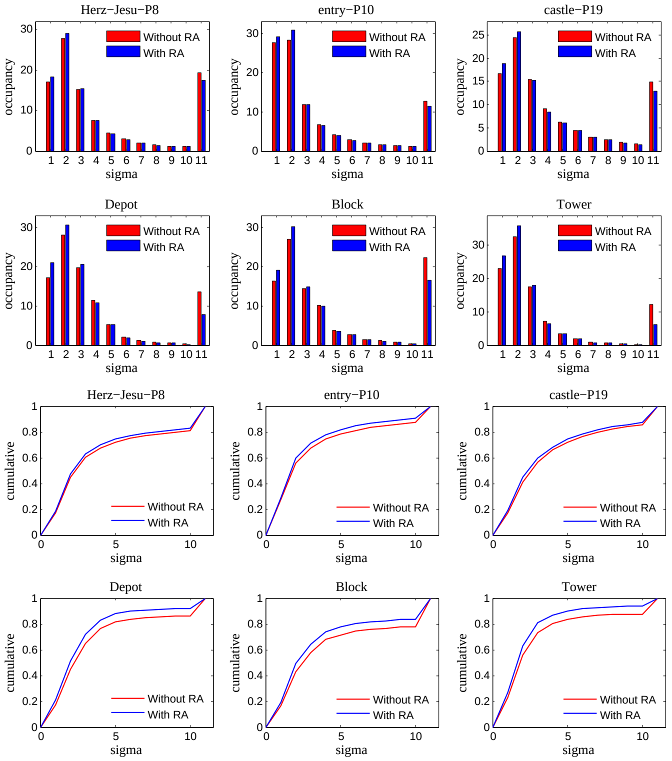


Figure 5. Relative error histograms and cumulative relative error curves of the real datasets with ground truth and the synthetic datasets. The measurements clearly confirm the improvement in absolute 3D accuracy after readjustment. Moreover, as the scales of the datasets increase, the improvement in absolute 3D accuracy correspondingly becomes more obvious.

Castle_20M datasets.

Finally, we use three synthetic datasets, Depot, Block and Tower, for further quantitative confirmation. We have also tested the performance of our method using different types and levels of artificial perturbations.

5.3. Evaluation

Since our method intends to re-optimize camera geometry after SfM and before dense reconstruction, it is independent of the dense reconstruction method, and highly applicable to almost any of them. That means the improvements of camera geometry will certainly benefit the subsequent dense reconstruction, no matter what dense reconstruction method we use. In this paper, we use the revised quasi-dense approach [19], as an example to demonstrate the good performance of our readjustment method.

Real datasets with ground truth Figure 5 shows the histograms built over the relative errors and curves upon the cumulative relative errors on the castle-P19, Herz-Jesu-P25, and castle-P30 datasets. Although the images in [25] are all captured in a well-conditioned environment, and the scales of the datasets are comparatively small, meaning these current existing real datasets with ground truth are not the ideal target for our readjustment approach, the measurements on all datasets still clearly demonstrate obvious improvements in absolute 3D accuracy after readjustment.

Dataset		Station	Casa	Street	Louvre			Castle		
					26	53	101	5.3M	10.5M	21M
Epipolar	Before RA	0.99	1.06	1.03	0.85	0.89	1.25	0.69	0.90	0.97
error	After RA	0.63	0.64	0.59	0.54	0.53	0.48	0.37	0.47	0.49
[pixels]	Reduction	36.4%	39.6%	42.7%	36.5%	40.4%	61.6%	46.4%	47.8%	49.5%

Table 2. The epipolar error of the real datasets without ground truth before and after readjustment.

Real datasets without ground truth Due to each of these datasets containing greater numbers of images with higher resolution, the improvements are more remarkable on these datasets. Without ground truth 3D data, we use the epipolar error as an alternative [9, 15] to give quantitative evaluations for the accuracy of local camera parameters. From Table 2, we note that the reduction rate of the epipolar error after readjustment ranges from 36.4% in the Station dataset to 61.6% in the Louvre_101 dataset. We also demonstrate that as the number and resolution of images increase, the improvement in the accuracy of local camera parameters become more remarkable. More concretely, the accuracy of local camera parameters increases more remarkably in the Louvre_101 dataset than the Louvre_53 and Louvre_26 datasets. Likewise, the Castle_21M dataset shows greater improvements in accuracy than the Castle_10M and Castle_5M datasets.

Figure 1 and Figure 6 shows some results using the method described in [19] to reconstruct dense patches and the method described in [17] to convert them into 3D mesh models. Obviously, our proposed method can effectively recover the delicate geometric structure, especially for large-scale high-resolution datasets, in which severe error propagation and estimation biases are inevitable.

Synthetic datasets Finally, the relative error histograms and cumulative relative error curves of the synthetic datasets are also shown in Figure 5. One important observation is that the improvement in absolute 3D accuracy on the smaller scale datasets, namely Herz-Jesu-P8, entry-P10, and castle-P19 are inferior to the improvement on datasets of a larger scale, namely Tower, Depot, Block datasets, which coincides with the experimental results on the real datasets without ground truth.

An interesting experiment is to perform the evaluation on the large synthetic datasets with ground truth, where additional perturbations are artificially introduced to quantify the improvement in absolute 3D accuracy using our method against different types and levels of noise (shown in Table 3). For uniform errors, we artificially add Gaussian noise to the parameters of all cameras, and for concentrated errors, we add Gaussian noise to the parameters of two specific cameras so that the average reprojection error after global bundle adjustment reaches different lev-

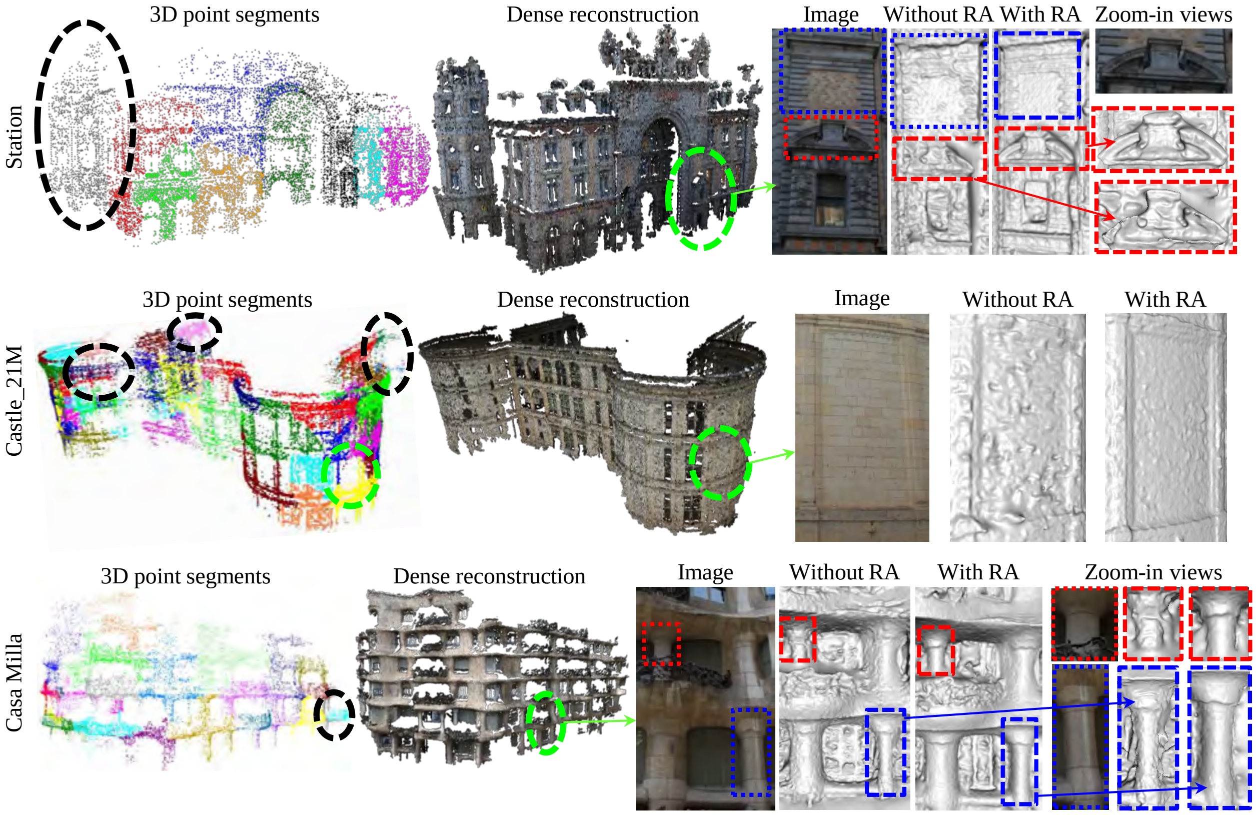


Figure 6. The comparison between mesh models with and without readjustment. A set of patches reconstructed by the method described in [19] and mesh models based on these patches are used to assess the accuracy of 3D structure and camera parameters. The images on the left are 3D point segmentation results and black dashed circles mark the invalid segments (because of insufficient 3D points or low accuracy). We note from the mesh models that severe errors and outliers are attenuated and detailed geometry is recovered by the readjustment (RA) approach.

Error type		Uniform error				Concentrated error			
Error level		1 pixel	2 pixels	5 pixels	20 pixels	1 pixel	2 pixels	5 pixels	20 pixels
Relative	Without RA	3.642	7.134	18.611	34.147	3.412	6.934	16.134	32.201
error	With RA	3.026	5.864	16.762	33.487	2.762	4.960	9.880	14.791
[sigma]	Reduction	16.91%	17.80%	9.94%	1.93%	19.05%	28.47%	38.76%	44.75%

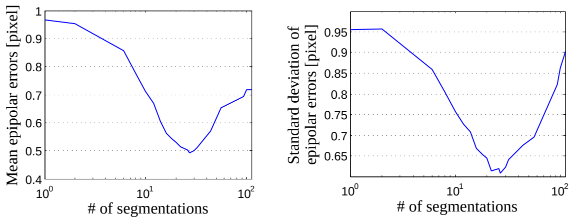
Table 3. The average relative error of the Block dataset with and without readjustment where different types and levels of perturbations are manually introduced.

els in terms of pixels. From Table 3, we can see that our method improves the absolute 3D accuracy on both types of errors, while the improvement in the datasets with concentrated errors is more remarkable, which means the local readjustment mechanism can effectively prevent any errors of poorly conditioned regions from being propagated to other regions.

Segmentation numbers and constraints Another experiment is to test the relationship between the mean and standard deviations of epipolar errors and the number of segments. We select the Castle_101 dataset as an example (Figure 7). We observe that as the number of segments increases, both the mean and standard deviations of epipolar errors first decrease and then increase. Therefore, oversegmentation may sometimes even lead to degeneracies in the 3D structure and camera parameters. However, one important observation is that the acceptable number of segments is in a large range. For example, in the Castle_101 dataset, the mean epipolar error is below 0.6 pixel when the number of segments ranges from approximately 10 to 50. We also note from Table 1 that the number of invalid segments in every dataset is very small. Therefore, our segmentation algorithm can readily converge into an acceptable segmentation result.

Figure 7 also uses a table to demonstrate the importance of the constraints of our segmentation algorithm. Note that when a constraint is not taken into consideration, its corresponding affinity measure in the 3D graph of the segmentation algorithm is also ignored. We observe that a segmentation method with both accuracy and density constraints performs the best. Therefore, both accuracy and density constraints can help generate well-conditioned segments for segment-based bundle adjustment.

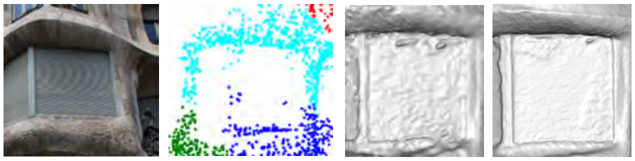
Segment fusion analysis As shown in Figure 8, we note that although the reconstructed region is at the junction of three segments, the inconsistencies in the geometry are neg-



Constraint	S+D+A	S+D	S+A	S	Origin
Mean epipolar error [pixels]	0.49	0.69	0.54	0.74	0.97
Standard deviations of epipolar error [pixels]	0.60	0.72	0.67	0.73	0.96

S: size constraint D: density constraint A: accuracy constraint.

Figure 7. Top: The mean and standard deviations of epipolar errors for different choices of the number of segments in the readjustment approach for the Castle_101 dataset. Bottom: The mean and standard deviations of epipolar errors for different choices of constraints in 3D point segmentation for the Castle_101 dataset.



(a) A sample image (b) Point seg. results (c) Without RA (d) RA+Fusion

Dataset		H-P8	e-P10	c-P19	Dept	Block	Tower
Average relative error [sigma]	Global	4.277	3.635	4.174	3.279	4.046	2.862
	Boudary	4.285	3.630	4.199	3.266	4.124	2.855

H-P8: Herz-Jesu-P8 e-P10: entry-P10 c-P19: castle-P19

Figure 8. Top: The boundary fusion result of the Casa_Milla dataset. Bottom: The comparison of the average relative error between the global reconstruction scene and the boundary region.

ligible in the dense reconstruction result after readjustment and boundary fusion. We can also conclude from the table in Figure 8 that the relative error in the boundary region after fusion is almost the same as the global ones, sometimes even smaller.

Running time Finally, we consider the running time of our system (see Table 1). We note that 3D point segmentation dominates the running time of our whole system, but only takes less than two minutes at most (the Castle_21M dataset). This is because we use [5], which eliminates the need for eigenvector computation, to optimize Normalized Cuts and that our inputs are sparse 3D points rather than dense ones. Moreover, since our inputs have been globally optimized, segment-based bundle adjustment converges quickly. Therefore, our approach is highly computationally efficient and a parallel implementation will further speed up this process.

6. Conclusion

In this paper, we propose a re-optimization method that partitions globally optimized sparse SfM points into well-conditioned segments for re-optimization, which can be fused back into a consistent model. The key to our approach is the introduction of local cameras. Our method, which is complementary to existing bundle adjustment algorithms, can remarkably improve the accuracy of 3D structure and camera geometry in addition to recovering detailed geometry especially for large-scale datasets.

References

[1] S. Agarwal, K. Mierle, and Others. Ceres solver. `https://code.google.com/p/ceres-solver/`.

[2] M. Aubry, K. Kolev, B. Goldluecke, and D. Cremers. Decoupling photometry and geometry in dense variational camera calibration. *In ICCV*, 2011.

[3] C. Bailer, M. Finckh, and H. P. A. Lensch. Scale robust multi view stereo. *In ECCV*, 2012.

[4] N. D. F. Campbell, G. Vogiatzis, C. Hernández, and R. Cipolla. Using multiple hypotheses to improve depth-maps for multi-view stereo. *In ECCV*, 2008.

[5] I. S. Dhillon, Y. Guan, and B. Kulis. Weighted graph cuts without eigenvectors: A multilevel approach. *PAMI*, 29(1):1944–1957, 2007.

[6] T. Fang and L. Quan. Resampling structure from motion. *In ECCV*, 2010.

[7] M. Farenzena, A. Fusiello, and R. Gherardi. Structure-and-motion pipeline on a hierarchical cluster tree. *In ICCV Workshop on 3D Digital Imaging and Modeling*, 2009.

[8] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski. Towards internet-scale multi-view stereo. *In CVPR*, 2010.

[9] Y. Furukawa and J. Ponce. Accurate camera calibration from multi-view stereo and bundle adjustment. *In CVPR*, 2008.

[10] Y. Furukawa and J. Ponce. Accurate, dense, and robust multi-view stereopsis. *PAMI*, 32(8):1362–1376, 2010.

[11] D. Gallup, J.-M. Frahm, P. Mordohai, and M. Pollefeys. Variable baseline/resolution stereo. *In CVPR*, 2008.

[12] R. Gherardi, M. Farenzena, and A. Fusiello. Improving the efficiency of hierarchical structure-and-motion. *In CVPR*, 2010.

[13] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz. Multi-view stereo for community photo collections. *In ICCV*, 2007.

[14] B. Goldlücke and D. Cremers. A super-resolution framework for high-accuracy multiview reconstruction. *In DAGM*, 2009.

[15] R. Hartley and A. Zisserman. Multiple view geometry in computer vision. *Cambridge University Press*, 2000.

[16] C. Hernández, F. Schmitt, and R. Cipolla. Silhouette coherence for camera calibration under circular motion. *PAMI*, 29(2):343–349, 2007.

[17] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. *In Symp. Geom. Proc.*, 2006.

[18] J.-M. Lavest, M. Viala, and M. Dhome. Do we really need an accurate calibration pattern to achieve a reliable camera calibration? *In ECCV*, 1998.

[19] M. Lhuillier and L. Quan. A quasi-dense approach to surface reconstruction from uncalibrated images. *PAMI*, 27(3):418–433, 2005.

[20] D. D. Morris. Gauge freedoms and uncertainty modeling for three-dimensional computer vision. *PhD thesis, Carnegie Mellon University*, 2001.

[21] K. Ni, D. Steedly, and F. Dellaert. Out-of-core bundle adjustment for large-scale 3D reconstruction. *In ICCV*, 2007.

[22] G. Sibley, C. Mei, I. Reid, and P. Newman. Adaptive relative bundle adjustment. *In Robotics: Science and Systems*, 2009.

[23] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3D. *In SIGGRAPH*, 2006.

[24] D. Steedly, I. Essa, and F. Dellaert. Spectral partitioning for structure from motion. *In ICCV*, 2003.

[25] C. Strecha, W. von Hansen, L. V. Gool, P. Fua, and U. Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. *In CVPR*, 2008.

[26] B. Triggs, P. Mclauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment - a modern synthesis. In *Vision Algorithms: Theory and Practice, LNCS*, pages 298–375. Springer Verlag, 2000.

[27] R. Tyleček and R. Šára. Depth map fusion with camera position refinement. *In CVWW*, 2009.

[28] H.-H. Vu, P. Labatut, J.-P. Pons, and R. Keriven. High accuracy and visibility-consistent dense multiview stereo. *PAMI*, 34(5):889–901, 2012.

[29] K.-Y. K. Wong and R. Cipolla. Reconstruction of sculpture from its profiles with unknown camera positions. *IEEE Transactions on Image Processing*, 2004.