Multi-View Geometry Compression

Siyu Zhu, Tian Fang[1], Runze Zhang, Long Quan

The Hong Kong University of Science and Technology

Abstract. For large-scale and highly redundant photo collections, eliminating statistical redundancy in multi-view geometry is of great importance to efficient 3D reconstruction. Our approach takes the full set of images with initial calibration and recovered sparse 3D points as inputs, and obtains a subset of views that preserve the final reconstruction accuracy and completeness well. We first construct an image quality graph, in which each vertex represents an input image, and the problem is then to determine a connected sub-graph guaranteeing a consistent reconstruction and maximizing the accuracy and completeness of the final reconstruction. Unlike previous works, which only address the problem of efficient structure from motion (SfM), our technique is highly applicable to the whole reconstruction pipeline, and solves the problems of efficient bundle adjustment, multi-view stereo (MVS), and subsequent variational refinement.

1 Introduction

Multi-view geometry represents the intricate geometric relations between multiple views of a 3D scene [1]. Together with the SfM points, it can be optimized as a Maximum Likelihood Estimation, the process of which is called bundle adjustment, and we can also base on multi-view geometry for MVS and subsequent variational refinement. However, for large-scale, irregularly sampled, and highly redundant photo collections, such as Internet photos and video sequences, the statistical redundancy in multi-view geometry severely decreases the reconstruction efficiency and increases the model storage space and transmission capacity.

The overwhelming majority of existing methods generally solve the problem of efficient SfM by eliminating image matching redundancy [2–8], hierarchical decomposition [9–14] and subsampling [15–18]. However, these techniques do not fundamentally reduce the redundancy in multi-view geometry and are difficult to address the problems of efficient MVS and variational refinement. In this paper, we proposed *Multi-View Geometry Compression* (MVGC), which intends to subsample redundant cameras and is highly applicable to the whole reconstruction pipeline, that is it can handle the problems of efficient bundle adjustment, MVS and variational refinement.

Intuitively, our proposed method is based on the observation that a small subset of images is sufficient to guarantee a consistent final reconstruction while preserving its reconstruction accuracy and completeness well. We then formulate the compression problem as a graph simplification procedure. Based on the full set of images with initial calibration and SfM points, we construct an image quality graph, where each image is

[1] Tian Fang is the corresponding author.

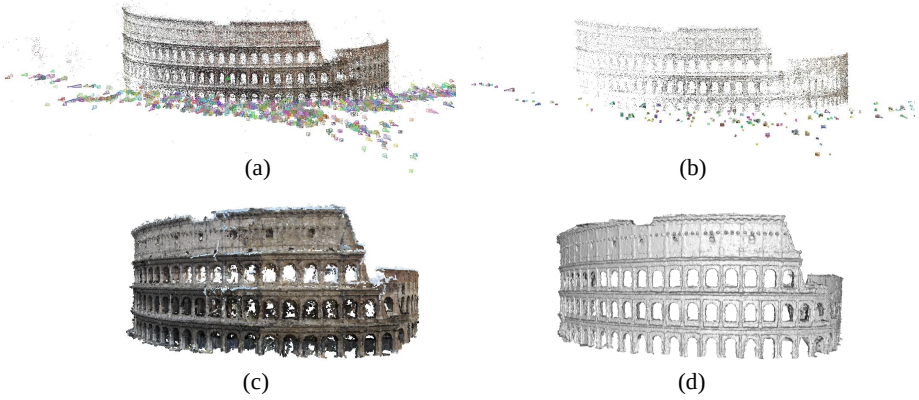


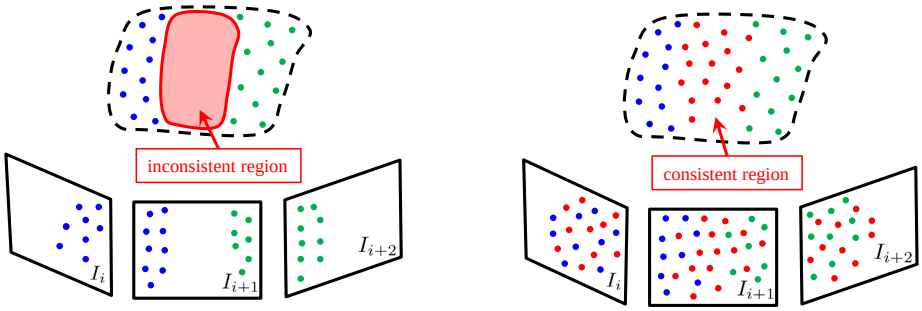
Fig. 1. Results of our method on the Colosseum data set containing 1789 images. (a) The initial full camera geometry and SfM points which are the inputs of our method. (b) The compressed camera geometry and SfM points that can be used for an efficient bundle adjustment. (c) The accurate and complete stereo reconstruction results based on the compressed multi-view geometry. (d) The variational refinement results from the compressed geometry. Note that it is too computationally expensive for the standard refinement approach to handle the full image collections.

a graph vertex and two vertexes are connected if their corresponding image pair can be used for a consistent reconstruction. The problem is then to determine a connected subgraph with the maximum sum of vertex weight under a certain number of vertexes. A down-sampling algorithm is then introduced as an approximation to solve the problem.

Our work is closely related to research on skeletal graph for efficient SfM [6]. The main difference is that our proposed approach is highly applicable to not only SfM, but also MVS and variational refinement applications. Moreover, the experiment results on both standard and Internet data sets also quantitatively and qualitatively demonstrate that our method increases the efficiency of bundle adjustment, MVS, and variational refinement by approximately a magnitude and better preserves the accuracy and completeness of the final reconstruction than the skeletal graph [6] and other methods.

2 Related Work

During the past few years, many works have been done to reduce redundancy in the 3D reconstruction pipeline. An intuitive idea is to eliminate unnecessary image matching pairs and the vast majority of previous works follow this direction. Rather than matching all image pairs, Agarwal et. al. [2] identify a small fraction of candidates by means of vocabulary tree recognition [3]. Frahm et al. [4] utilize approximate GPS tags to capture only nearby image pairs for matching. Wu [5] exploits the characteristic of large-scale matching problem that most of image pairs fail to match, and introduce the preemptive feature matching to filter out superfluous image matching pairs. Since we assume that image matching is given and initial multi-view geometry is also recovered in this paper, all the techniques mentioned are complementary to our work.



(a) Inconsistent reconstruction (b) Consistent reconstruction

Fig. 2. The interpretation for a consistent reconstruction. (a) and (b) show that, to preserve a consistent reconstruction, especially for MVS and variational refinement, the intersection of common visible points between the image pair $\{I_i, I_{i+1}\}$ and $\{I_{i+1}, I_{i+2}\}$ cannot be empty.

The proposed algorithm in this paper is closely related to [2, 6–8], which obtain a small skeletal subset of images from a dense scene graphs. However, rather than directly discarding redundant cameras as we do, these works only intend to accelerate the bundle adjustment process by simplifying matching between image pairs and cannot be appropriately applied to other reconstruction steps (e.g. MVS and variational refinement). Moreover, these methods fail to strictly guarantee that the simplified geometry can be used for a consistent reconstruction, which is properly handled in our work by the introduction of quality graph. The hierarchical framework [15] which proposes to resample a dense SfM points still suffers the same problems. Likewise, the approaches above are complementary to our work.

Similar to our method, the authors of [16–18] also propose subsampling approaches to solve SfM problems of video sequences, while they cannot handle diverse and unordered images with complex geometry topology. Canonical view selection methods for robot localization [19] also refer to similar criteria, however, they need different considerations when applied to reconstruction problems.

Instead of reducing input redundancy, divide-and-conquer is also widely used to solve large-scale reconstruction problems. Steedly et al. [9] utilize spectral partitioning to decouple original sequential images into pieces for easier bundle adjustment. Ni et al. [10] split the entire bundle problem into sub-maps with their own coordinate systems for efficient operation, and the authors of [11, 12, 20] also handle the bundle problem within a relative coordinate system. Rather than recover 3D structure hierarchically at the image level, Farenzena et al. [13] and Gherardi et al. [14] apply divide-and-conquer at the variable level.

3 Problem Formulation

The input to our approach is a set of images $\mathbf{I} = \{I_i \,|\, i = 1, 2, ..., N_\mathbf{I}\}$, where $N_\mathbf{I}$ is the number of cameras. Their corresponding camera calibrations are denoted by $\mathbf{\Pi} =$

$\{\Pi_i | i = 1, 2, ..., N_{\mathbf{I}}\}$, which are obtained from a standard reconstruction pipeline [21]. A point cloud $\mathbf{P} = \{P_m \,|\, j = 1, 2, ..., N_{\mathbf{P}}\}$ is then obtained with triangulation, where $N_{\mathbf{P}}$ is the number of 3D points. It is noteworthy that $\mathbf{P}$ can either be sparse 3D points generated by a general SfM system (e.g. Bundler [22]), or quasi-dense points obtained from a commonly used MVS pipeline (e.g. Quasi-Dense [23]).

Now, the problem of MVGC is defined as follows: given an image set $\mathbf{I}$, its corresponding camera geometry $\mathbf{\Pi}$, and triangulated point cloud $\mathbf{P}$, find an image subset $\mathbf{I}'$ that yields a reconstruction with the least loss of reconstruction accuracy and completeness, which are measured by an increasing function $w(I_i)$ for a given image I_i.

Another issue with our approach is to guarantee that the compressed multi-view geometry can be used for a consistent reconstruction (see Fig. 2), and we therefore introduce the *image quality graph* $G_{\mathbf{I}}$, where $G_{\mathbf{I}} = (\mathbf{V}, \mathbf{E})$ is an undirected connected graph, with $\mathbf{V} = \{V_i \,|\, i = 1, 2, ..., n\}$, and each graph vertex V_i corresponds to an image I_i. In order to guarantee that I_i and I_j can be used for a consistent reconstruction, there ought to exist at least one image I_k satisfying that

$$(\mathbf{P}_{I_i} \cap \mathbf{P}_{I_k}) \cap (\mathbf{P}_{I_j} \cap \mathbf{P}_{I_k}) \neq \emptyset, \tag{1}$$

where $\mathbf{P}_{I_i}$ is the set of 3D points visible in camera I_i. And for the graph edge $E_{ij} \in \mathbf{E}$ connecting vertex V_i and V_j, its edge weight $h(E_{ij})$ is defined as the number of cameras $\{I_k\}$ satisfying equation (1), that is $h(E_{ij}) = |\{I_k\}|$. Obviously, V_i and V_j is disconnected when $h(E_{ij}) = 0$.

In summary, the problem of MVGC can be mathematically formulated as: given $(\mathbf{I}, \mathbf{\Pi}, \mathbf{P})$ and N_I which is the target number of cameras to be preserved, find $\mathbf{I}' \subset \mathbf{I}$ satisfying

$$\begin{aligned} &\mathbf{I}' = \arg\max_{\mathbf{I}'} \sum_{I_i \in \mathbf{I}'} w(I_i), \\ &s.t. \quad |\mathbf{I}'| = N_I \ \text{ and } \ G_{\mathbf{I}'} \text{ is connected.} \end{aligned} \tag{2}$$

4 Approaches

4.1 Graph Construction

Recalling that $G_{\mathbf{I}} = (\mathbf{V}, \mathbf{E})$, and each graph vertex V_i has a weight scale $w(I_i)$, which is an increasing function denoting the possibility I_i tends to be preserved in the final subgraph. Obviously, we prefer cameras with high accuracy and completeness measures and we further introduce the regularity measure to avoid irregularly sampled views in the reconstruction scene. Therefore, the weight scale $w(I_i)$ of image I_i is defined as

$$w(I_i) = w_a(I_i) \cdot w_c(I_i) \cdot w_r(I_i), \tag{3}$$

where $w_a(I_i)$ is the accuracy measure, $w_c(I_i)$ the completeness measure, and $w_r(I_i)$ the regularity measure. Please see Fig. 3 for the visual demonstration.

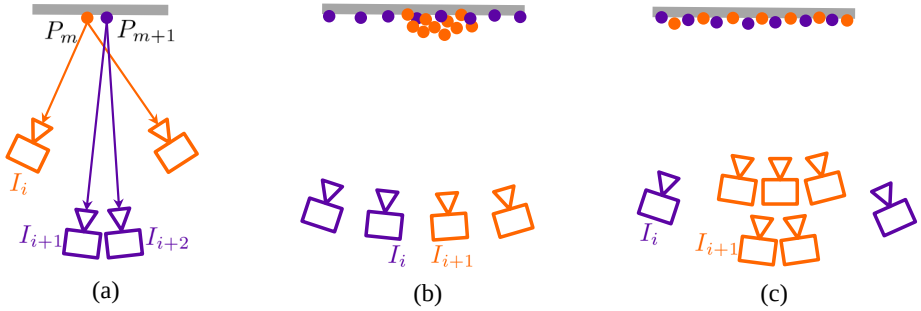


Fig. 3. The illustration of graph vertex measures. (a) 3D points triangulated by cameras with large angles and pixel sampling rate are generally of low covariance (e.g. P_m over P_{m+1}), and cameras with great numbers of such points tend to be remained. (b) Cameras covering 3D points of low density are more inclined to preserve reconstruction completeness (e.g. I_i over I_{i+1}). (c) We prefer under-sampled cameras rather than over-sampled cameras (e.g. I_i over I_{i+1}) to guarantee a uniform distribution of cameras.

Accuracy measure. First, the remained cameras should preserve the final reconstruction accuracy as much as possible. Without ground truth data, camera covariance can be regarded as an alternative to assess the reconstruction accuracy, while even the fast gauge-free covariance estimation method [23] is extremely both space and time consuming for large-scale image collections.

Geometrically, cameras covering more well-qualified 3D points tend to be better constrained and of smaller covariance. For one camera, we utilize the sum of accuracy measure of its visible 3D points as an approximation of camera covariance. Inspired by [24, 25], camera pairs with large angles and pixel sampling rate generally triangulate 3D points of high quality (see Fig. 3(a)), and the accuracy measure of a single point P_m observed by the image pair $\{I_i, I_j\}$ is therefore defined as

$$\begin{aligned} g_a(P_m, I_i, I_j) &= e^{\left(-\frac{(\angle I_i P_m I_j - \sigma_\theta)^2}{2\sigma^2_{\angle I_i P_m I_j}}\right)} \cdot s(P_m, I_i, I_j) \\ s.t. \quad \sigma_{\angle I_i P_m I_j} &= \begin{cases} \theta_1 & \angle I_i P_m I_j \le \sigma_\theta \\ \theta_2 & \angle I_i P_m I_j > \sigma_\theta \end{cases} \end{aligned}, \tag{4}$$

where $s(P_m, I_i, I_j) = \min(1/r(P_m, I_i), 1/r(P_m, I_j))$, $r(P_m, I_i)$ quantifies the diameter of a sphere centered at P_m and its projected diameter equals the pixel spacing in I_i, $\sigma_\theta = 20°$, $\theta_1 = 5°$, and $\theta_2 = 15°$.

Therefore, the accuracy measure of one camera I_i is expressed as

$$w_a(I_i) = \sum_{\substack{I_j \in \mathbf{I},\ i \neq j \\ P_m \in \mathbf{P}_{I_i} \cap \mathbf{P}_{I_j}}} g_a(\angle I_i P_m I_j). \tag{5}$$

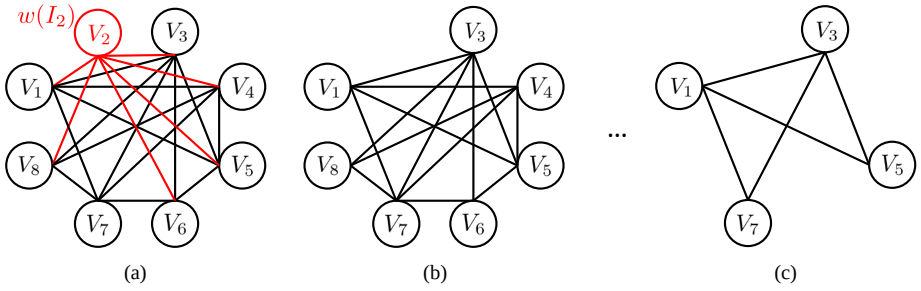


Fig. 4. The graph simplification algorithm. Starting from the initial dense graph shown in (a), we iteratively remove the vertex with the lowest weight and its corresponding adjacent edges until the number of vertexes below a threshold (the final graph is shown in (c)). Meanwhile, the obtained sub-graph is connected so as to guarantee a consistent model.

Completeness measure. Next, the preserved cameras should cover the whole reconstructed scene, and the completeness in 3D space can be measured by the density of points in 3D. Generally, cameras containing more 3D points of low density better preserve reconstruction completeness (see Fig. 3(b)), and we compute the sum of N_c lowest point density as our camera completeness measure, namely

$$w_c(I_i) = \max_{\mathbf{P}'_{I_i} \subseteq \mathbf{P}_{I_i}, |\mathbf{P}'_{I_i}| = N_c} \sum_{P_m \in \mathbf{P}'_{I_i}} d(P_m), \tag{6}$$

where $d(P_i)$ is the inverse of the density of P_i, and $N_c = 10$.

Regularity measure. Finally, we should guarantee that the selected cameras are regularly sampled in the reconstruction scene rather than over-sampled in some regions with popular viewpoints and under-sampled in others. Similar to [24], we take the scene content, appearance, and scale into consideration, and introduce the regularity measure $w_r(I_i)$ an increasing function computing the degree the points covered by I_i are also covered by its adjacent cameras. Quantitatively, each point is given a weight counteracting a greater number of visible cameras with a good range of parallax within a neighborhood, and $w_r(I_i)$ is given as the sum of its visible point weights, that is

$$w_r(I_i) = \sum_{\substack{I_j \in \mathbf{I},\ i \neq j \\ P_m \in \mathbf{P}_{I_i} \cap \mathbf{P}_{I_j}}} g_N(P_m) \cdot g_S(P_m). \tag{7}$$

Here $g_N(P_m)$ penalizes the trend of greater numbers of features in common with a decreasing angle, and $g_N(P_m)$ is given as

$$g_N(P_m) = \prod_{\substack{I_i, I_j \in \mathbf{I},\ i \neq j \\ P_m \in \mathbf{P}_{I_i} \cap \mathbf{P}_{I_j}}} g_r(P_m, I_i, I_j), \tag{8}$$

Table 1. Statistics of the SfM data sets and algorithms.

		Sequential			Unstructured		
		Garden	Park	Street	Colosseum	Notre Dame	Trevi
# of images		948	940	684	1789	712	1789
# of images after MVGC		95	94	68	179	71	179
Mean position error [m]	Key frame	0.31	0.24	0.26	–	–	–
	Skeletal graph	0.93	0.82	0.89	0.41	0.34	0.43
	MVGC	0.24	0.19	0.22	0.12	0.11	0.09
Running time [min]	MVGC	0.94	0.88	0.42	2.88	1.57	3.11
	BA with MVGC	0.49	0.66	0.49	0.91	0.45	0.87
	BA without MVGC	6.84	8.14	3.30	10.22	7.23	12.01

where $g_r(P_m, I_i, I_j) = \min((\alpha/\alpha_{\max})^2, 1)$, α is the angle between the rays of I_i and I_j triangulating P_m, and we set $\alpha_{\max} = 60^\circ$, which prefers camera pairs with large angles in all our experiments.

Moreover, $g_S(P_m)$ encourages views with equal or higher resolution than the reference view, and we use

$$g_S(P_m) = \begin{cases} 2/r & 2 \leq r \\ 1 & 1 \leq r < 2 \\ r & r < 1 \end{cases} \tag{9}$$

where $r = r(P_m, I_i)/r(P_m, I_j)$.

4.2 Graph Simplification

Unfortunately, the problem of computing the optimal sub-graph is NP-complete, and we propose an approximation algorithm to obtain it. Starting from an initial dense graph, we first sort the graph vertexes into a priority queue based on their weight and then iteratively delete the vertex with the lowest weight and its adjacent edges until the number of vertexes equals the target number. Meanwhile, there are two issues we should take into consideration. First, the graph edge weight, namely $h(E_{ij})$, changes along with the vertex removal process. When $h(E_{ij}) = 0$, V_i and V_j is disconnected, and we ought to guarantee that the image quality graph is connected until the convergence is met. Second, when the graph is simplified, the graph vertex weight $w(I_i)$ also varies. Obviously, it is computationally unnecessary to update the vertex weight of the entire graph each time a vertex is removed. In our implementation, we update the vertex weight and reallocate the priority queue each time 10 vertexes are removed. Admittedly, the obtained sub-graph is not theoretically optimal, but it still generates satisfactory results.

5 Experiments

To demonstrate that the proposed MVGC is highly applicable to the whole pipeline of 3D reconstruction, we present the applications of our method in bundle adjustment, MVS, and variational refinement in the following experiments.

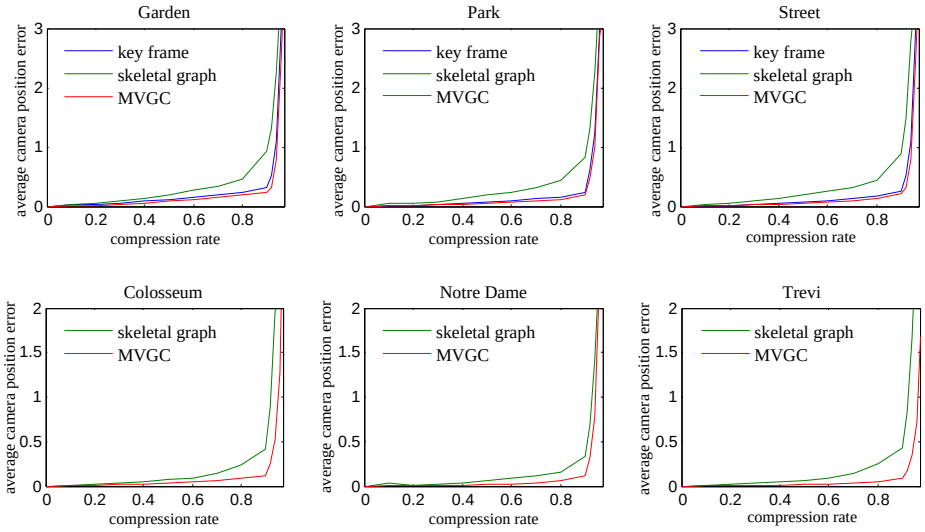


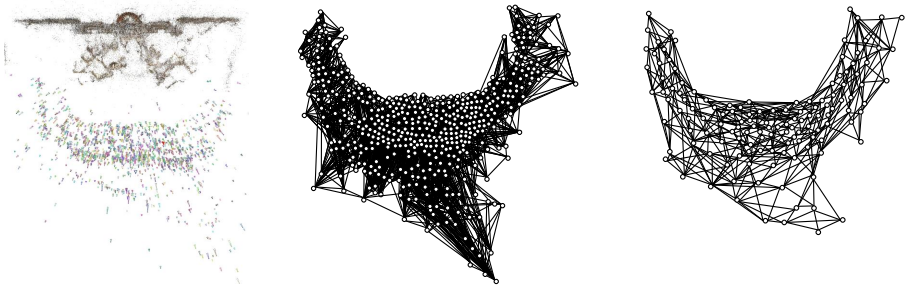
Fig. 5. Average camera position errors after bundle adjustment at different compression rates. We can see that the results of MVGC have lower average camera position errors than those using key frame and skeletal graph on all data sets.

Implementation Our approach is implemented in C++ and the code is completely CPU-based. All our experiments are tested on a PC with a Quad-core Intel 3.5GHz processor and 32GB RAM. We use the publicly available Ceres [26] for bundle adjustment and Quasi-Dense approach [23] for MVS. We also follow the pipeline described in [27] for mesh generation and [28] for variational refinement. The parameters in our implementation are all standard except for N_c, which determines the density of the quality graph and directly effects the running time of MVGC.

Data set Three categories of data sets are introduced for testing, and they are respectively video sequences with a resolution of 3M pixels obtained from monochrome video cameras, unstructured high-resolution images from a well-known benchmark [29], and unstructured photo collections from the Internet captured by various cameras with different focal lengths, distortion and sensor noise, under different conditions of lighting, surface reflection and scale.

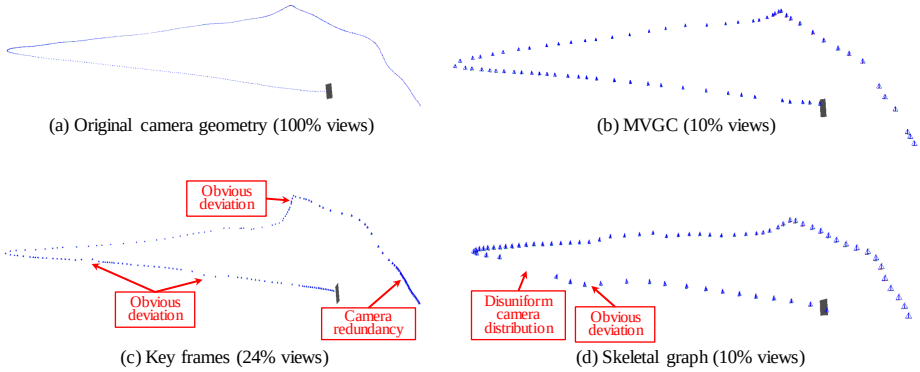
5.1 Applications in Bundle Adjustment

First, the input images are automatically calibrated using the standard approach [21] and MVGC is then introduced to select a subset of initially calibrated views for bundle adjustment. In our implementation, we avoid using multiple full bundle adjustment, which is the main time-consuming part of SfM, before MVGC but divide a large SfM problem into small sub-problems, which are partially bundled, and use the partially



(a) Original geometry (b) Original image quality graph (c) Simplified image quality graph

Fig. 6. Graph simplification results of the Trevi data set. Note how the MVGC algorithm preserves the important topology, but significantly reduces the redundancy of the original graph.



(a) Original camera geometry (100% views) (b) MVGC (10% views)

(c) Key frames (24% views) (d) Skeletal graph (10% views)

Fig. 7. Comparisons of video sequence trajectories after bundle adjustment. Compared with the original full sequence, MVGC best preserves the video sequence trajectory after bundle adjustment while the resampled video sequence trajectories using other methods have obvious deviation, redundancy, or disuniform distribution.

bundled geometry as the input geometry for MVGC. After that the full bundle adjustment of each sub-problem and the final bundle adjustment of the global problem are performed. In details, we incrementally add cameras and do partial bundle adjustment for the recently 10 added cameras. When the number of cameras increases relatively by a certain ratio (e.g. 10%), the geometry of the newly added cameras is regarded as the initially calibrated geometry, and we use MVGC to resample these cameras. Next, the newly resampled cameras together with previously resampled cameras are used for a full bundle adjustment. Then, MVGC is introduced again to resample all the remained cameras and a full bundle adjustment is performed finally. In other words, MVGC do require camera calibrations, but only rough camera calibrations.

The author in [5] shows that partial BA guarantees good camera geometry locally, which can provide satisfactory input geometry for MVGC. The author in [5] also proves

Table 2. Statistics of the MVS data sets and algorithms.

		Benchmark data sets		Internet data sets		
		fountain-R25	Herz-Jesu-R23	Square	Notre Dame	Colosseum
# of images		25	23	1907	712	1789
# of images after MVGC		13	12	191	71	179
Running time [min]	MVGC	0.13	0.15	3.26	1.57	2.88
	MVS with MVGC	6.44	6.98	90.14	33.13	85.01
	MVS without MVGC	17.48	16.20	922.23	344.80	835.60

that for the majority of large-scale dataset, the time cost of partial BA is much smaller than those of full BA. So the efficiency of SfM improves greatly as we remarkably improve the efficiency of full BA as shown in Table 1. For example, the overall time of full BA of Trevi dataset with and without MVGC are 0.87 and 12.01 minutes respectively. The partial BA and the other time costs with and without MVGC are 2.73 and 3.96 minutes respectively. Therefore, the overall time of SfM with and without MVGC are 3.6 and 15.97 minutes respectively.

Table 1 also shows the statistics of the data sets for bundle adjustment, and about 90% of cameras are removed using the method of key frame, skeletal graph [6], and MVGC respectively. Here we define compression rate as the percentage of discarding cameras. We can see from Table 1 and Fig. 5 that, on both sequential and unstructured data sets, MVGC obviously outperforms key frame and skeletal graph in average camera position errors after bundle adjustment while improving the efficiency of bundle adjustment by almost a magnitude. Fig. 7 gives the qualitative demonstration of the outstanding performance of our method. Fig. 6 also provides the visual demonstration of the graph simplification result of the Trevi data set.

5.2 Applications in Multi-view Stereo

Both on the benchmark and Internet data sets, we use full image sets for SfM and MVGC to select a subset of views for dense 3D reconstruction. The general statistics of MVS data sets shown in Table 2 indicate that approximately 50% of cameras are discarded in the benchmark data sets and 90% in the Internet data sets. As shown in Fig. 8(a) and Fig. 8(b), the relative error of MVS reconstruction with MVGC is almost the same as that using full image sets on the benchmark data sets. The curves of average relative errors at different compression rates of the Internet data sets are provided in Fig. 8, which further confirms the superiority of MVGC over the skeletal graph and key frame in applications of MVS. Fig. 9 provides visual results of MVS of the fountain-R25 and Notre Dame data sets. Unlike the MVS results using the skeletal graph and random image selection, which are compromised by incomplete regions, our compressed multi-view geometry preserves reconstruction accuracy and completeness best in the final reconstruction.

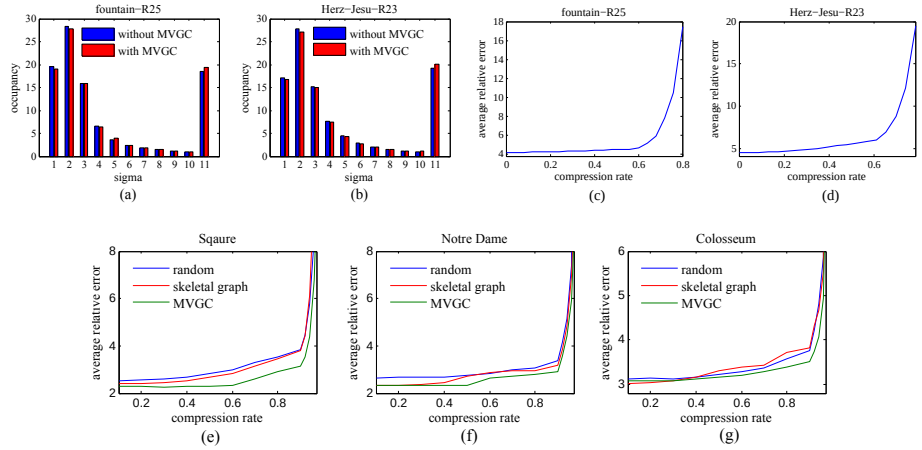


Fig. 8. Average relative errors of benchmark and Internet data sets in the MVS application. (a) and (b) are comparisons of average relative errors of the benchmark data sets between reconstruction using full image sequences and compressed geometry (50% views). We observe that the absolute reconstruction accuracy using compressed geometry is almost the same as that using complete geometry. (c) and (d) show the average relative errors of the benchmark data sets at different compression rates. (e), (f) and (g) present the average relative errors at different compression rate of the Internet data sets. Likewise, MVGC outperforms the skeletal graph and random image selection methods in any compression rate.

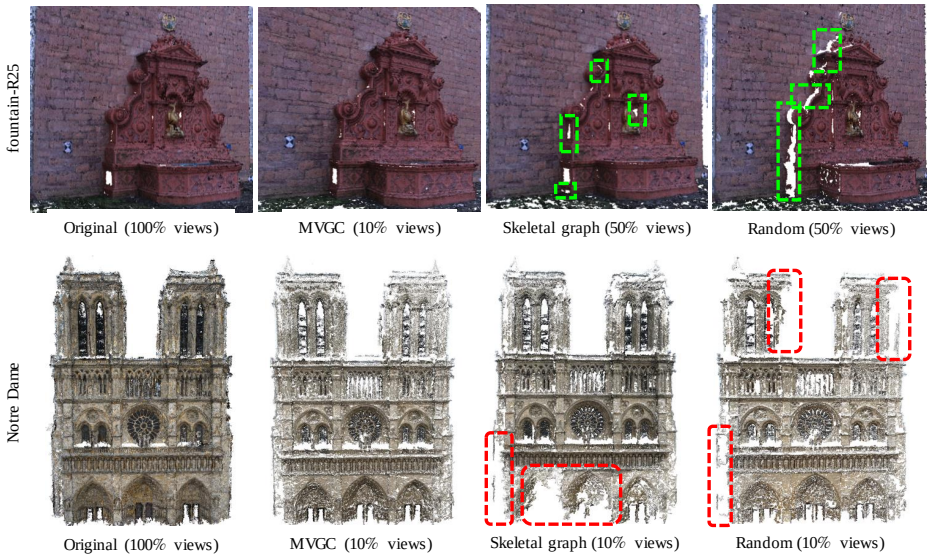


Fig. 9. Comparisons of MVS dense results of the fountain-R25 and Notre Dame data sets in the MVS application. Note the incomplete regions highlighted by dashed rectangles and that the reconstruction from the compressed geometry is complete.

Table 3. Statistics of variational refinement data sets and algorithms.

		Benchmark data sets			Internet data sets		
		fountain-P11	entry-P10	castle-P19	Trevi	Square	Basilica
# of images		11	10	19	1789	1907	1103
# of images after MVGC		6	5	10	54	57	33
Running time [min]	MVGC	0.09	0.07	0.12	3.88	3.67	2.98
	Refinement with MVGC	12.01	8.23	22.61	273.13	324.67	160.00
	Refinement without MVGC	43.36	36.93	86.04	N/A	N/A	N/A

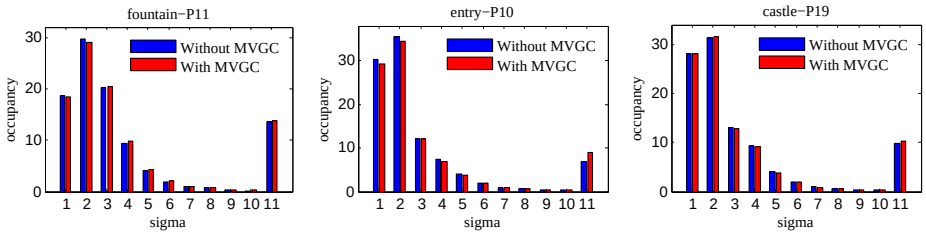


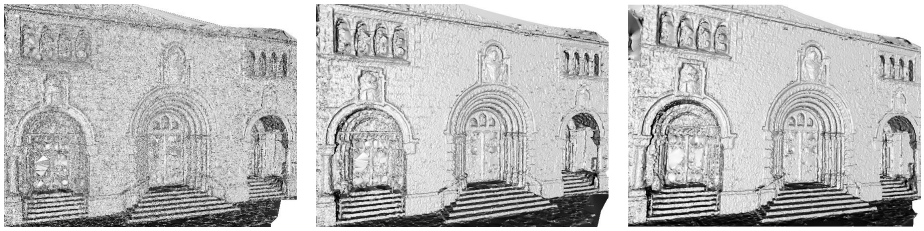
Fig. 10. Relative error histograms of the benchmark data sets in the variational refinement application. It is obvious that the remarkably improved efficiency using MVGC almost compromises no reconstruction accuracy.

5.3 Applications in Variational Refinement

In this application, the full image sets are used for SfM and MVS, and we then introduce MVGC to select a subset of views for variational refinement. According to the statistics in Table 3, the proposed algorithm reduces approximately 50% of the images in the benchmark data sets and 97% of the images in the Internet data sets for refinement. As demonstrated by the relative error histograms in Fig. 11, the reconstruction accuracy using compressed geometry is almost the same as the one using full image sets on the benchmark data sets but we only spend about 25% of the running time of the method without compression. Since standard refinement algorithms are generally based on an energy summation traversing all image pairs to compute the cross correlation of photo consistency, as the number of images explodes, the increased time consumption becomes unacceptable, particularly for redundant Internet data sets. Take the Trevi data set as an example, for the full image set approximately 319k image pairs are needed for pairwise correlation computation, while only 3% of these image pairs are needed after MVGC. Fig. 11 and Fig. 12 provide some visual results of variational refinement of the benchmark and Internet data sets.

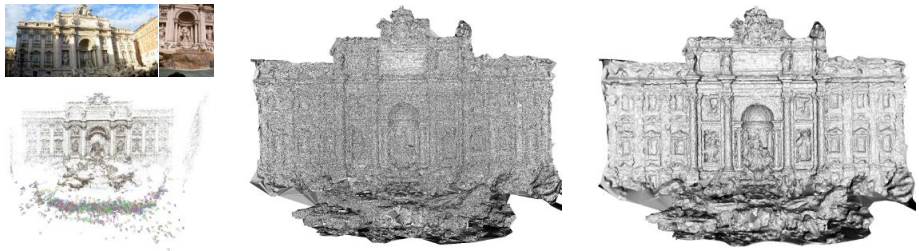
5.4 Discussions

Graph vertex weight measures Another important issue for our experiments is to validate the effectiveness of the accuracy, completeness, and regularity measures in the

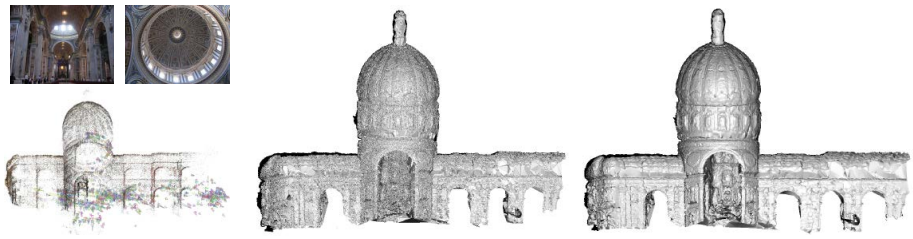


(a) The original triangulated mesh (b) 100% views for refinement (c) 32% views for refinement using MVGC

Fig. 11. Comparisons of variational refinement results of the Herz-Jesu-P25 data set in the variational refinement application. The refinement results using compressed geometry still recovers the same details, edges, and topology as the refinement results using full image sets.



(a) Trevi Fountain Reconstruction (3% views for refinement)



(b) St. Peter's Square Reconstruction (3% views)

Fig. 12. Variational refinement results of the Internet data sets using MVGC. The figures from left to right are samples of Internet image collections, SfM points and corresponding camera geometry, initial triangulated meshes with a decent amount of noise, and finally refined mesh models. We should note that the data sets above cannot be handled by standard refinement methods using full image sets.

image quality graph. Table 4 shows the average camera position errors of the sequential and unstructured data sets in SfM for different choices of vertex weight measures. We observe that the MVGC algorithm with accuracy, completeness, and regularity measures performs the best while the absence of anyone of these weight measures will lead to the degeneration of the reconstruction accuracy.

Table 4. Average position errors for different choices of vertex weight measures after bundle adjustment. "A" is the accuracy measure, "C" the completeness measure, and "R" the regularity measure. The MVGC algorithm with accuracy, completeness, and uniformity measures has the lowest average position error after bundle adjustment.

		A+C	A+R	C+R	A+C+R
Sequential data sets	Garden	0.32	0.27	0.30	0.24
	Park	0.26	0.20	0.24	0.19
	Street	0.34	0.31	0.29	0.22
Unstructured data sets	Colosseum	0.19	0.18	0.16	0.12
	Notro Dame	0.22	0.16	0.18	0.11
	Trevi	0.14	0.12	0.12	0.09

Running time As indicated in Table 1, Table 2, and Table 3, the running time of MVGC is minor, and the largest is 3.67 minutes for the Trevi data set. While our proposed method significantly reduces the time and memory consumption of bundle adjustment, MVS, and variational refinement by almost a magnitude, and makes the previously impossible variational refinement using Internet data sets manageable.

6 Conclusions

We propose an approach to compress multi-view geometry by obtaining a subset of views from the original full camera geometry while maximizing the reconstruction accuracy and completeness. The key technical contribution is the introduction of image quality graph, and the problem is then transformed to computing a sub-graph from the original dense graph. MVGC is highly applicable to bundle adjustment, MVS, and variational refinement, and introduces remarkable improvement in efficiency with almost no loss of reconstruction accuracy and completeness. The experiment results on both standard and Internet data sets demonstrate the remarkable improvement in efficiency with almost no loss of reconstruction accuracy and completeness. As for the future work, it is interesting to theoretically explore the optimum subset of multi-view geometry, although our approximation approach has provided satisfactory results. Going forward, hopefully we could extend our method to a hierarchical pipeline so as to validate our pipeline on larger data sets, especially the most prevalent city-scale data sets.

Acknowledgement. We really appreciate the support of RGC-GRF 618711, RGC/NSFC N_HKUST607/11, ITC-PSKL12EG02, and National Basic Research Program of China (2012CB316300).

References

1. Heyden, A., Pollefeys, M.: Tutorial on multiple view geometry. In conjunction with ICPR (2000)
2. Agarwal, S., Snavely, N., Simon, I., Seitz, S.M., Szeliski, R.: Building rome in a day. In ICCV (2009)

3. Nister, D., Stewenius, H.: Scalable recognition with a vocabulary tree. In CVPR (2006)
4. Frahm, J., Georgel, P.F., Gallup, D., Johnson, T., Raguram, R., Wu, C., Jen, Y., Dunn, E., Clipp, B., Lazebnik, S., Pollefeys, M.: Building rome on a cloudless day. In ECCV (2010)
5. Wu, C.: Towards linear-time incremental structure from motion. In 3DTV (2013)
6. Snavely, N., Seitz, S.M., Szeliski, R.: Skeletal sets for efficient structure from motion. In CVPR (2008)
7. Li, X., Wu, C., Zach, C., Lazebnik, S., Frahm., J.: Modeling and recognition of landmark image collections using iconic scene graphs. In ECCV (2008)
8. Agarwal, S., Snavely, N., Seitz, S.M., Szeliski, R.: Bundle adjustment in the large. In ECCV (2010)
9. Steedly, D., Essa, I., Dellaert, F.: Spectral partitioning for structure from motion. In ICCV (2003)
10. Ni, K., Steedly, D., Dellaert, F.: Out-of-core bundle adjustment for large-scale 3D reconstruction. In ICCV (2007)
11. Mouragnon, E., Lhuillier, M., Dhome, M., Dekeyser, F., Sayd, P.: Real time localization and 3D reconstruction. In CVPR (2006)
12. Eudes, A., Lhuillier, M.: Error propagations for local bundle adjustment. In CVPR (2009)
13. Farenzena, M., Fusiello, A., Gherardi, R.: Structure-and-motion pipeline on a hierarchical cluster tree. In ICCV Workshop on 3D Digital Imaging and Modeling (2009)
14. Gherardi, R., Farenzena, M., Fusiello, A.: Improving the efficiency of hierarchical structure-and-motion. In CVPR (2010)
15. Fang, T., Quan, L.: Resampling structure from motion. In ECCV (2010)
16. Fitzgibbon, A.W., Zisserman, A.: Automatic camera recovery for closed or open image sequences. In ECCV (1998)
17. Nistér, D.: Reconstruction from uncalibrated sequences with a hierarchy of trifocal tensors. In ECCV (2000)
18. Repko, J., Pollefeys, M.: 3D models from extended uncalibrated video sequences. In Proc. 3DIM (2005)
19. Booij, O., Zivkovic, Z., Krose, B.: Sparse appearance based modeling for robot localization. In IROS (2006)
20. Zhu, S., Fang, T., Xiao, J., Quan, L.: Local readjustment for high-resolution 3d reconstruction (2014)
21. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge University Press (2000)
22. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3D. In SIGGRAPH (2006)
23. Lhuillier, M., Quan, L.: A quasi-dense approach to surface reconstruction from uncalibrated images. PAMI **27** (2005) 418–433
24. Goesele, M., Snavely, N., Curless, B., Hoppe, H., Seitz, S.M.: Multi-view stereo for community photo collections. In ICCV (2007)
25. Furukawa, Y., Curless, B., Seitz, S.M., Szeliski, R.: Towards internet-scale multi-view stereo. In CVPR (2010)
26. Agarwal, S., Mierle, K., Others: Ceres solver. (https://code.google.com/p/ceres-solver/)
27. Kazhdan, M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. In Symp. Geom. Proc. (2006)
28. Delaunoy, A., Prados, E., Gargallo, P., Pons, J.P., Sturm, P.F.: Minimizing the multi-view stereo reprojection error for triangular surface meshes. In BMVC (2008)
29. Strecha, C., Hansen, W.V., Gool, L.V., Fua, P., Thoennessen, U.: On benchmarking camera calibration and multi-view stereo for high resolution imagery. In CVPR (2008)