Supplementary Material of Stochastic Bundle Adjustment for Efficient and Scalable 3D Reconstruction

Lei Zhou[1][0000−0003−4988−5084], Zixin Luo[1][0000−0001−6946−2826], Mingmin Zhen[1][0000−0002−8180−1023], Tianwei Shen[1][0000−0002−3290−2258], Shiwei Li[2][0000−0003−0712−0059], Zhuofei Huang[1][0000−0002−8479−7568], Tian Fang[2][0000−0002−5871−3455], and Long Quan[1][0000−0001−8148−1771]

[1] Hong Kong University of Science and Technology
{lzhouai,zluoag,mzhen,tshenaa,zhuangbr,quan}@cse.ust.hk
[2] Everest Innovation Technology
{sli,fangtian}@altizure.com

1 Overview

The supplementary material is organized as below.

– In Sec. 2, we analyze how the time and space complexity are reduced by solving the split reduced camera system (RCS) with our STBA in place of the original RCS.
– In Sec. 3, we present ablation studies on the stochastic graph clustering algorithm proposed in Sec. 4.4 of the main paper and the effect of the maximum cluster size.
– In Sec. 4, the full algorithm of STBA is laid out.

2 Complexity Analysis

In this section, we will analyze how our STBA reduces the time and space complexity by solving the split reduced camera system (RCS) (Eqs. 13 of the main paper) in place of the original RCS (Eq. 5 of the main paper), whether the exact or inexact linear solver is used, as shown in Table 1. Please note that the analysis considers the most general case and does not presuppose any special structures, *e.g.*, the extreme sparsity, of the RCS.

Table 1: **The time and space complexity of LM and our STBA when solving the reduced camera system.** m denotes the camera number. Γ is the maximum cluster size. κ and κ' are the condition numbers of the Schur complement and split Schur complement after preconditioning. r and r' denote the edge number and the sampled edge number of the camera graph, respectively.

	Cholesky factorization		Conjugate gradient	
	LM	STBA	LM	STBA
Time complexity	$O(m^3)$	$O(m\Gamma^2)$	$O(r\sqrt{\kappa})$	$O(r'\sqrt{\kappa'})$
Space complexity	$O(m^2)$	$O(m\Gamma)$	$O(r)$	$O(r')$

Cholesky factorization is known to have a cubic time complexity and a quadratic space complexity in the camera number m when solving the RCS [3]. If using Cholesky factorization to solve the split RCS exactly, the time and space complexity of each sub-problem of STBA are $O(\Gamma^3)$ and $O(\Gamma^2)$, respectively, where Γ is the maximum cluster size. Since there are $O(m/\Gamma)$ sub-problems, the time and space complexity of STBA are $O(m\Gamma^2)$ and $O(m\Gamma)$, respectively. With Γ being a constant, the time and space complexity of STBA are linear with m.

Besides the exact solvers, conjugate gradient is an inexact approach to solving the linear equations iteratively. It is known to have a $O(r\sqrt{\kappa})$ time complexity and a $O(r)$ space complexity when solving the RCS [11], where r is the camera connection number and κ is the condition number of the Schur complement $\mathbf{S}$ (see Eq. 5 of the main paper). However, $\mathbf{S}$ is generally ill-conditioned, which necessitates preconditioning to reduce the condition number κ [3, 9, 5, 8]. The amount of decrease in κ depends on how accurately preconditioning can be performed. If using conjugate gradient to solve the split RCS inexactly, STBA reduces the time complexity to $O(r'\sqrt{\kappa'})$ and the space complexity to $O(r')$. Here, r' is the sampled camera connection number, and κ' is the maximum condition number of the split Schur complements $\{\mathbf{S}_i\}_{i=1}^{l}$ (see Eqs. 10 of the main paper) after preconditioning. Due to the sampling of the camera connections, r' is smaller than r. In our experiments, r' is less than one fifth of r when we set the maximum cluster size Γ to 100. The condition number κ' also should be smaller than κ, because preconditioning the low-dimensional $\mathbf{S}_i$ can be performed more accurately and efficiently than the high-dimensional $\mathbf{S}$.

3 Ablation Studies on Stochastic Graph Clustering

In Sec. 4.4 of the main paper, we have proposed a stochastic graph clustering (SGC) algorithm to sample the chance constraints in each iteration. In this section, we would like to conduct ablation studies on the clustering strategies and the maximum cluster size Γ. First, we make comparisons with 3 clustering methods below.

- **KMeans** which partitions the camera centers into k clusters by using the K-Means algorithm. In order to introduce randomness, we randomly choose k camera centers as the initial means in the first step.
- **NCut** which uses normalized cut for graph clustering as in the previous works [10, 14, 13]. We turn the camera graph into a random one by keeping its edges with the probability proportional to the edge weights and then run normalized cut on it.
- **NSGC** is the abbreviation for **non-**stochastic graph clustering. It is a variant of SGC which uses the classic greedy Louvain's algorithm [4] rather than joining clusters randomly as SGC.

Apart from the clustering strategies, we run all the algorithms with 6 different maximum cluster sizes which are $\Gamma = 1$, 25, 50, 100, 200 and ∞. Here, "$\Gamma = 1$"

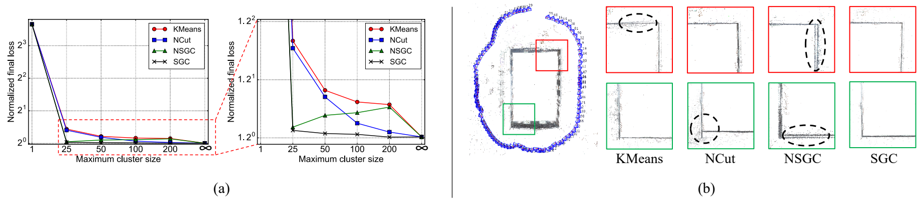

Fig. 1: (a) **Normalized final losses w.r.t. the maximum cluster size** produced by different clustering methods. The smaller the loss is, the better convergence is attained. (b) **Reconstructions of Gerrard Hall from the COLMAP dataset [1].** All the methods except our SGC lead to layered facade reconstruction results, as marked by the dashed circles.

means that each camera forms a cluster. And "$\Gamma = \infty$" means all the cameras are grouped into a single cluster, in which case STBA is equivalent to the classic LM algorithm without using clustering. We run all the methods on each problem of 1DSfM [12] and KITTI [7] in the same way as Sec. 5.2 of the main paper. We record the final losses of all the clustering algorithms and normalize them with the division by the minimum loss that the algorithms attained. Therefore, the smaller the normalized loss is, the better convergence is achieved. We show the average normalized losses of different methods in Fig. 1(a).

First of all, the proposed SGC reaches the minimum losses at all the cluster sizes, showing its efficacy compared with KMeans and NCut. The disadvantage of KMeans is that it does not utilize the camera connectivity for clustering, as opposed to NCut and SGC. In comparison with SGC, NCut partitions a graph into clusters in a top-down manner. The downside of this strategy is that it does not explicitly decide whether an edge at the bottom level will be selected or discarded with a defined probability as SGC does (see Eq. 17 of the main paper). Since NCut always stops once the cluster sizes are smaller than Γ, some nodes may constantly stay in the same clusters without being exposed to the cuts. Instead, the bottom-up strategy of SGC considers the selection of every edge from the very beginning and contributes to the better convergence than NCut in the end. Besides, the outperformance of SGC over NSGC indicates the necessity of making the graph clustering randomized for better convergence.

Second, all of KMeans, NCut and SGC have better convergence as Γ increases. It is reasonable because the larger Γ is, the more chance constraints can be sampled, leading to a more accurate approximation by chance constrained relaxation in Sec. 4.2 of the main paper. In the extreme case when $\Gamma = 1$, all the chance constraints are neglected ($i.e.$, the confidence level $\alpha = 0$ in Eq. 12 of the main paper). It induces poor approximations for the STBA iterations and hence leads to very bad convergence. However, the final loss can be reduced by an order of magnitude by just increasing Γ to 25. Besides, it is noteworthy that SGC is the least sensitive to Γ compared against NCut and KMeans, as the loss does not vary a lot when Γ changes from 25 to 200. Different from other methods, NSGC gets the larger loss when Γ increases from 25 to 200. We found that it is because NSGC uses fixed clusters and neglects the geometric constraints between the clusters all the time, which would cause the inconsistency between the geometries of different clusters. The problem is more severe when the cluster size increases, as it is less flexible to align large clusters seamlessly than small clusters. And the reduced flexibility of large clusters is more likely to cause layered geometries at the cluster boundaries (see Fig. 1(b)). In Fig. 1(b), we show the reconstruction results of *Gerrard Hall* from the COLMAP dataset [1] produced by different graph clustering methods with $\Gamma = 50$. All the methods except our SGC lead to layered facade reconstruction results.

4 Full Algorithm

Below we lay out the full algorithm of STBA for reference.

Algorithm 1: Stochastic Bundle Adjustment (STBA)

Input: Visibility graph: $\mathcal{G} = (\mathcal{C} \cup \mathcal{P}, \mathcal{E})$, initial pose and point parameters: $\mathbf{x} = [\mathbf{c}^T \mathbf{p}^T]^T = \mathbf{x}_0$

Output: $\mathbf{x}^*$ minimizing $F(\mathbf{x})$

$t = 0$, $t_{max} = 100$, $\lambda = 1e-4$, $\Gamma = 100$, $\epsilon = 1e-6$, stop=False

Build camera graph $\mathcal{G}_c = (\mathcal{C}, \mathcal{E}_c)$ from $\mathcal{G}$

while *(not stop) and* $t++ < t_{max}$ **do**

  `/* Stochastic graph clustering */`

  $\{\Phi_i\}_{i=1}^{l} = StochasticGraphClustering(\mathcal{G}_c, \Gamma)$ `//` Γ `is the maximum cluster size`

  Build the equality constraint matrix $\mathbf{A}$ according to $\{\Phi_i\}_{i=1}^{l}$ `// see Eq. 8 of the main paper`

  `/* Evaluations */`

  Evaluate reprojection errors $\mathbf{f}$ and Jacobian $\mathbf{J_c}, \mathbf{J_p}, \mathbf{J}'_\mathbf{p}, \mathbf{J}' = [\mathbf{J_c}, \mathbf{J}'_\mathbf{p}]$

  $\mathbf{C} = \mathbf{J_p}^T \mathbf{J_p} + \lambda \mathrm{diag}(\mathbf{J_p}^T \mathbf{J_p})$, $\mathbf{E} = \mathbf{J_c}^T \mathbf{J_p}$, $\mathbf{w} = \mathbf{J_p}^T \mathbf{f}$

  $\mathbf{B} = \mathbf{J_c}^T \mathbf{J_c} + \lambda \mathrm{diag}(\mathbf{J_c}^T \mathbf{J_c})$

  $\mathbf{C}' = \mathbf{J}'_\mathbf{p}{}^T \mathbf{J}'_\mathbf{p} + \lambda \mathrm{diag}(\mathbf{J}'_\mathbf{p}{}^T \mathbf{J}'_\mathbf{p})$

  $\mathbf{E}' = \mathbf{J_c}^T \mathbf{J}'_\mathbf{p}$

  $\mathbf{g} = -\mathbf{J}'\mathbf{f}$

  `/* Steepest descent correction */`

  if $\lambda \geq 0.1$ **then**

    $\mathbf{H}_\lambda = \mathbf{J}'^T \mathbf{J}' + \lambda \mathbf{D}'^T \mathbf{D}'$

    $\tilde{\mathbf{H}}_\lambda = \mathrm{diag}(\mathbf{H}_\lambda)$

    $\boldsymbol{\nu} = (\mathbf{A} \tilde{\mathbf{H}}_\lambda^{-1} \mathbf{A}^T)^{-1} \mathbf{A} \tilde{\mathbf{H}}_\lambda^{-1} \mathbf{g}$

    $\mathbf{g} = \mathbf{g} - \mathbf{A}^T \boldsymbol{\nu}$

  $\mathbf{g} \triangleq [\mathbf{v}'^T \mathbf{w}'^T]^T$

  $\mathbf{S}' = \mathbf{B} - \mathbf{E}' \mathbf{C}'^{-1} \mathbf{E}'^T \triangleq \{\mathbf{S}_i\}_{i=1}^{l}$

  $\mathbf{b}' = \mathbf{v}' - \mathbf{E}' \mathbf{C}'^{-1} \mathbf{w}' \triangleq \{\mathbf{b}_i\}_{i=1}^{l}$

  `/* Solve pose steps in parallel */`

  for $i = 1$ *to* l **do**

    Solve $\mathbf{S}_i \Delta \mathbf{c}_i = \mathbf{b}_i$

  $\Delta \mathbf{c} = [\Delta \mathbf{c}_1^T ... \Delta \mathbf{c}_l^T]^T$

  $\Delta \mathbf{p} = \mathbf{C}^{-1}(\mathbf{w} - \mathbf{E}^T \Delta \mathbf{c})$

  $\mathbf{x} = [\mathbf{c}^T \mathbf{p}^T]^T$, $\Delta \mathbf{x} = [\Delta \mathbf{c}^T \Delta \mathbf{p}^T]^T$

  if *(Cost tolerance* $< \epsilon$*) or (Gradient tolerance* $< \epsilon$*) or (Parameter tolerance* $< \epsilon$*) (see [2])* **then**

    stop=True

  if $F(\mathbf{x}) > F(\mathbf{x} + \Delta \mathbf{x})$ **then**

    $\lambda = \lambda/3$, $\mathbf{c} = \mathbf{c} + \Delta \mathbf{c}$, $\mathbf{p} = \mathbf{p} + \Delta \mathbf{p}$

  else

    $\lambda = \lambda * 3$

$\mathbf{x}^* = [\mathbf{c}^T \mathbf{p}^T]^T$

References

1. https://colmap.github.io/datasets.html
2. Agarwal, S., Mierle, K., Others: Ceres solver. http://ceres-solver.org
3. Agarwal, S., Snavely, N., Seitz, S.M., Szeliski, R.: Bundle adjustment in the large. In: ECCV (2010)
4. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of statistical mechanics: theory and experiment **2008**(10) (2008)
5. Dellaert, F., Carlson, J., Ila, V., Ni, K., Thorpe, C.E.: Subgraph-preconditioned conjugate gradients for large scale slam. In: IROS (2010)
6. Dolan, E.D., Moré, J.J.: Benchmarking optimization software with performance profiles. Mathematical programming **91**(2), 201–213 (2002)
7. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: CVPR (2012)
8. Jian, Y.D., Balcan, D.C., Dellaert, F.: Generalized subgraph preconditioners for large-scale bundle adjustment. In: Outdoor and Large-Scale Real-World Scene Analysis (2012)
9. Kushal, A., Agarwal, S.: Visibility based preconditioning for bundle adjustment. In: CVPR (2012)
10. Ni, K., Steedly, D., Dellaert, F.: Out-of-core bundle adjustment for large-scale 3d reconstruction. In: ICCV (2007)
11. Shewchuk, J.R., et al.: An introduction to the conjugate gradient method without the agonizing pain (1994)
12. Wilson, K., Snavely, N.: Robust global translations with 1dsfm. In: ECCV (2014)
13. Zhang, R., Zhu, S., Fang, T., Quan, L.: Distributed very large scale bundle adjustment by global camera consensus. In: ICCV (2017)
14. Zhu, S., Shen, T., Zhou, L., Zhang, R., Wang, J., Fang, T., Quan, L.: Parallel structure from motion from local increment to global averaging. arXiv preprint arXiv:1702.08601 (2017)