

# Propagation-based incremental triangulation for multiple views 3D reconstruction

Wei Fang (方维)<sup>1</sup>, Kui Yang (杨奎)<sup>2</sup>, and Haiyuan Li (李海源)<sup>1</sup>

<sup>1</sup>School of Automation, Beijing University of Posts and Telecommunications, Beijing 100876, China

<sup>2</sup>School of Instrumentation Science and Opto-electronics Engineering, Beihang University, Beijing 100191, China

\*Corresponding author: [fangwei@bupt.edu.cn](mailto:fangwei@bupt.edu.cn)

Received July 26, 2020 | Accepted September 11, 2020 | Posted Online December 9, 2020

To balance the accuracy and efficiency in multiple-view triangulation with sequential images, a high-efficiency propagation-based incremental triangulation (INT) method, carving three-dimensional (3D) scene points by updating the incoming feature track one by one without iterations, is proposed. Based on the INT method, a more accurate iteration-limited INT method is also established with few iterations to bound the propagated errors, ensuring the accuracy of subsequent 3D reconstruction. Finally, experimental results demonstrate that the proposed methods can balance the efficiency and accuracy in different multiple-view INT situations.

**Keywords:** multiple views; 3D reconstruction; incremental triangulation; propagation.

**DOI:** [10.3788/COL202119.021101](https://doi.org/10.3788/COL202119.021101)

## 1. Introduction

There has been an escalation in the amount of work dealing with three-dimensional (3D) reconstruction<sup>[1-3]</sup>, pose estimation<sup>[4-6]</sup>, structure-from-motion (SfM)<sup>[7,8]</sup>, and simultaneous localization and mapping (SLAM)<sup>[9,10]</sup>, such as applications in robotics, augmented/virtual reality, and self-driving. A fundamental component of these works is triangulation, which refers to recovering the 3D location from multiple two-dimensional (2D) image observations with known pose and camera parameters.

Over the years, many triangulation methods have been proposed, in particular for the case of two or three views, where robust and fast methods have been developed<sup>[11,12]</sup>. It is being verified that more observations of the same scene point can improve the estimated position, but this also increases the complexity of the problem<sup>[13]</sup>. Therefore, the procedure given an arbitrary number of views and noisy observations to perform triangulation is still a difficult task. One of the most well-known triangulation actions is the midpoint method<sup>[14]</sup>, choosing the midpoint of the common perpendiculars to all of the rays as the best position estimation, which is also considered as the fastest one given multiple views by far. Nevertheless, this method is not supposed to be optimal, and the final 3D localization will potentially be inaccurate due to noisy inputs.

For more accurate triangulation performance, the achievement of optimal 3D scene points is given enough attention, but most solve for very small numbers of views, such as two or three views<sup>[14]</sup>. For two views, a closed-form solution was

proposed by Hartley and Sturm<sup>[15]</sup> and made more efficient and simplified by Lindstrom<sup>[16]</sup>. These methods yield all stationary points of their cost function, which are then checked to obtain the optimal solution. For three views, the closed-form triangulation method was achieved<sup>[17]</sup>, and they make use of the Grobner basis to solve polynomial equations. Nevertheless, the computational complexity increases exponentially with the number of views, and there is no suitable polynomial solver for more than three views.

To ensure the global optimum, there are also some methods based on convex optimization on the  $L_\infty$ -norm cost function instead of the  $L_2$ -norm solution<sup>[18]</sup>. However, it is known that  $L_\infty$ -norm optimization is extremely vulnerable to outliers<sup>[19]</sup>. Although some outlier removal method is achieved for  $L_\infty$ -norm in multi-view geometry<sup>[20]</sup>, the computational cost of  $L_\infty$ -norm is considerable, and the convex programs have to be done at each iteration to determine the update direction<sup>[21]</sup>. Therefore, the  $L_2$ -norm method is still a popular alternative to balance the speed and accuracy for multiple-view triangulation.

Based on the  $L_2$ -norm methods, state-of-the-art incremental triangulation (INT) problems usually can be solved in two steps, where initialization is achieved within the first few views, and then the sequential image comes to the system one by one to establish a continuous multiple-view triangulation problem. An obvious solution is to collect all of the observations of the 3D point when encountering a coming observation, and then a new multiple-view triangulation action is performed repeatedly<sup>[22]</sup>. However, it is a time-consuming process to perform multiple-view triangulation with iterative optimization<sup>[23,24]</sup>,

especially in large-scale scenarios, where there are usually a significant number of scene points and cameras. The previous work<sup>[24]</sup> mainly paid attention to the triangulation issue after collecting all of the observational vectors at one time, lacking consideration of the INT and its convergence characteristics. Thus, instead of establishing an unpredictable iteration-based multiple-view cost function when encountering a new feature track one by one, a propagation-based INT method is proposed in this Letter to carve the 3D estimate only with the incoming observation.

## 2. Notations and Preliminaries

The midpoint method for triangulation has a clear geometric description, where the estimated 3D location is the midpoint of the common perpendicular to the two observation rays given a two-view triangulation. For more than two views, it also means the minimum distances from the 3D point to all of the observation vectors. As shown in Fig. 1, the location of the 3D point  $\mathbf{p}$  can be approached by minimizing the accumulation of the distance  $\|\mathbf{e}_i\|^2$ .

Thus, the 3D point  $\mathbf{p}$  is obtained by minimizing the sum of squares of the distance for the 3D point to all the  $N$  observation vectors:

$$e(\mathbf{p}) = \sum_{i=1}^N \|\mathbf{e}_i\|^2 = \sum_{i=1}^N \|\mathbf{B}_i(\mathbf{p} - \mathbf{o}_i)\|^2, \quad (1)$$

where  $\mathbf{o}_i$  is the camera center, and  $\mathbf{B}_i = \mathbf{I}_{3 \times 3} - \mathbf{v}_i \mathbf{v}_i^T$  is the projection distance transformation from the point  $\mathbf{p}$  to the observational vector  $\mathbf{v}_i$ .

## 3. Propagation-Based Incremental Triangulation

Although the midpoint method is fast enough for multiple-view triangulation, it is considered inaccurate when the camera views are nearly parallel. Moreover, as depicted in Fig. 1, when the 3D

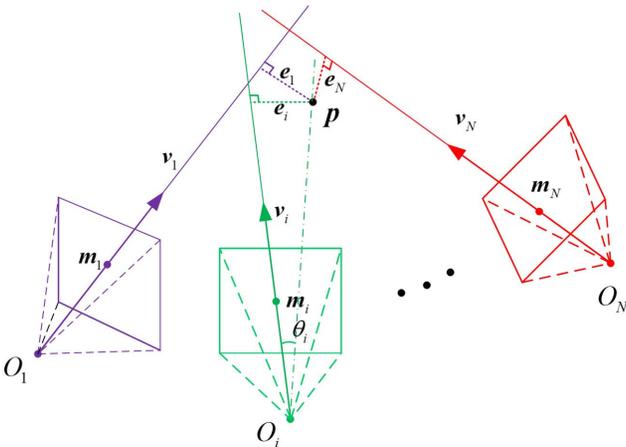


Fig. 1. Geometric indication of multi-view triangulation.

point  $\mathbf{p}$  moves far away from the camera center  $\mathbf{o}$ , the error  $\|\mathbf{e}\|$  increases. Nevertheless, all the 3D points that lay on the line  $\mathbf{o}\mathbf{p}$  coincide with the same observation on the 2D image plane. Therefore, the midpoint method is biased, and it is prone to overestimate the error for the 3D point that is far away from the camera center.

To address the biased estimation of the midpoint method by formulating their optimization problem in terms of the image reprojection error, the angular reprojection error is another popular choice, and it is supposed to own superior adaptability to different camera models<sup>[25]</sup>. Inspired by these angular reprojection errors methods, a weight that is proportional to the distance between the 3D point and the camera center is proposed. Thus, the unbiased temporal cost function for multiple-view triangulation is reformed as the following:

$$e(\mathbf{p}) = \sum_{i=1}^N \|\mathbf{e}_i(\mathbf{p})\|^2 = \sum_{i=1}^N \|\omega_i(\mathbf{p})\mathbf{B}_i(\mathbf{p} - \mathbf{o}_i)\|^2, \quad (2)$$

where an explicit weight is assigned by the inversely proportional function to the distance between the 3D point and the camera:

$$\omega_i(\mathbf{p}) = \frac{1}{\|\mathbf{p} - \mathbf{o}_i\|}. \quad (3)$$

According to Eq. (2) derived from the  $L_2$ -norm reprojection error, it is interesting to find that it is a close approximation to the angular error model, where

$$\sin \theta_i = \frac{\|\mathbf{e}_i\|}{\|\mathbf{o}\mathbf{p}\|} = \|\omega_i(\mathbf{p})\mathbf{B}_i(\mathbf{p} - \mathbf{o}_i)\|. \quad (4)$$

Thus, the cost function is unbiased for all of the points lying on the same observational vector, regardless of the depth about the 3D point to be estimated. Besides,  $\sin \theta_i$  is considered as a close approximation to exact angular error  $\theta_i$  (as shown in Fig. 1) when obtaining an accurate 3D point estimation. Compared to the optimization problems in terms of the  $L_2$ -norm image reprojection error, this optimization model is more robust for multiple triangulation when encountering fisheye or omnidirectional cameras.

To minimize the unbiased  $N$  view triangulation problem in Eq. (2), its gradient  $\nabla e(\mathbf{p})$  can be derived as

$$\nabla e(\mathbf{p}) = 2 \sum_{i=1}^N \{\omega_i^2(\mathbf{p})(\mathbf{p} - \mathbf{o}_i)^T [\mathbf{B}_i - \mathbf{e}_i(\mathbf{p})]\mathbf{I}\}. \quad (5)$$

Thus, the target 3D point  $\mathbf{p}^*$  is derived from  $\nabla e(\mathbf{p}^*) = 0$ , resulting in the following equation:

$$\left[ \sum_{i=1}^N \omega_i^2(\mathbf{p}^*) \mathbf{B}_i \right] \mathbf{p}^* = \sum_{i=1}^N [\omega_i^2(\mathbf{p}^*) \mathbf{B}_i \mathbf{o}_i + \omega_i^2(\mathbf{p}^*) e_i(\mathbf{p}^*) (\mathbf{p}^* - \mathbf{o}_i)]. \quad (6)$$

In most SfM or SLAM applications, it is well known that temporal triangulation is a repeated work when new 2D observations are extracted from incoming images. More exactly, the 3D scene point may be triangulated in the first few images, and then these available 3D points would be applied to estimate the pose of a new coming image by perspective-n-point (PnP)<sup>[26]</sup>, followed by more scene points with triangulation operations repeatedly. During these processes, INT is carried out with the feature track lasting for many frames with the same 3D point.

To address the large-scale and sequential image situations, an efficient INT is expected for multiple-view reconstruction based on the initial 3D point estimation. Fortunately, based on the initial estimation in Eq. (1), the next 3D point  $\mathbf{p}_n$  can be propagated based on the former  $\mathbf{p}_{n-1}$  when encountering sequential feature tracks without iteration, where

$$\mathbf{p}_n = \left[ \sum_{i=1}^N \omega_i^2(\mathbf{p}_{n-1}) \mathbf{B}_i \right]^{-1} \sum_{i=1}^N [\omega_i^2(\mathbf{p}_{n-1}) \mathbf{B}_i \mathbf{o}_i + \omega_i^2(\mathbf{p}_{n-1}) e_i(\mathbf{p}_{n-1}) (\mathbf{p}_{n-1} - \mathbf{o}_i)]. \quad (7)$$

It has been confirmed that the achievable reconstruction error decays quadratically as more views are added to the system<sup>[13]</sup>, and thus INT would improve the accuracy of the 3D scene with more observations. Besides, to bound the accumulation of propagation error during INT, an iteration-limited INT (ININT) is also proposed to achieve a more accurate 3D location estimation based on the propagated result. As depicted in Eq. (8),  $\mathbf{p}_{\text{ININT}}$  is derived from the INT result  $\mathbf{p}_{\text{INT}}$ , and the iteration operations continue until the given tolerance  $\epsilon_{\text{ININT}}$  is satisfied:

$$\begin{cases} \mathbf{p}_{\text{ININT}} = \left[ \sum_{i=1}^N \omega_i^2(\mathbf{p}_{\text{INT}}) \mathbf{B}_i \right]^{-1} \sum_{i=1}^N [\omega_i^2(\mathbf{p}_{\text{INT}}) \mathbf{B}_i \mathbf{o}_i + \omega_i^2(\mathbf{p}_{\text{INT}}) e_i(\mathbf{p}_{\text{INT}}) (\mathbf{p}_{\text{INT}} - \mathbf{o}_i)] \\ \|\mathbf{p}_{\text{ININT}} - \mathbf{p}_{\text{INT}}\| \leq \epsilon_{\text{ININT}} \end{cases} \quad (8)$$

In addition to the subsequent INT, the propagation-based theme is applied when more observations of the same 3D point arrived based on a reliable initial estimation. Increasing observational information will further optimize the existing triangulation results<sup>[13]</sup>. Therefore, the proposed INT (ININT) method inherits the convergence property of the unbiased midpoint triangulation method by propagating the current image observation.

## 4. Experiments

Experimental results are presented to validate the accuracy and efficiency of the proposed methods. The other four benchmarks

of multiple triangulation methods are compared. The first one is the native multiple-view middle point (MVMP) method; it is currently the fastest method for multiple-view triangulation, though it is labeled as inaccurate. The second method is the unbiased midpoint method (IRMP)<sup>[24]</sup>. The third one is considered a benchmark by minimizing the cost function of Eq. (2) using the naive Newton (NN) method. The fourth method is the gradient-based minimization of the reprojection errors, which is abbreviated as the standard gradient minimization of the reprojection errors (GMREs)<sup>[14]</sup> and popularly used in the recent state-of-the-art SLAM system. All of the experiments are carried out using MATLAB on a single thread on an Intel i7-8550U CPU at 1.80 GHz with 8G memory.

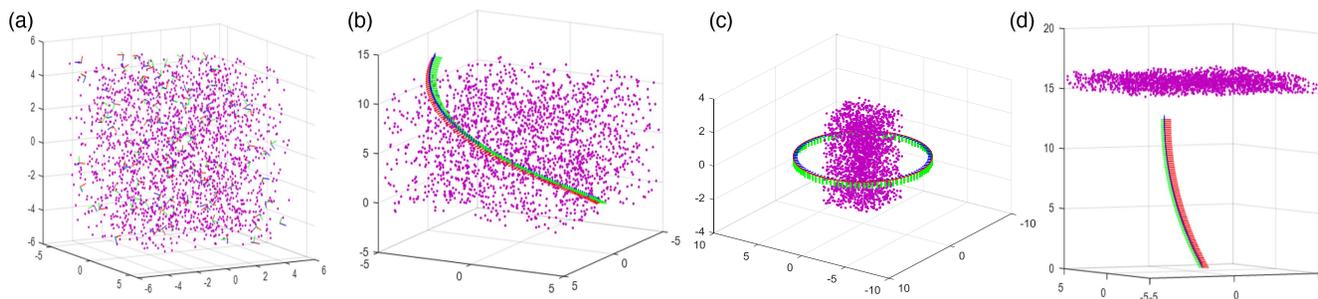
### 4.1. Experiments on synthetic data

In the synthetic dataset, the same camera calibration parameters are used with an image size of  $1024 \times 1024$  pixels and a focal length of 400 pixels. Four scenarios for multiple-view INT are simulated.

- Type A: The cameras and the 3D points are randomly distributed, as shown in Fig. 2(a). This synthesizes large-scale 3D reconstruction with images from different localizations.
- Type B: The camera moves along a curved trajectory with continuous triangulation, as shown in Fig. 2(b). It can simulate a robot that explores freely with an unpredicted landmark.
- Type C: The camera moves on a circle around the 3D points, as shown in Fig. 2(c). This can be seen as 3D reconstruction with a rotating platform.
- Type D: The camera moves along a curved trajectory towards the 3D scene, as shown in Fig. 2(d). This simulates the INT when the robot moves toward a target.

As illustrated in Fig. 2, 100 cameras and 1000 3D points are generated as different synthetic scenarios. Then, these 3D points are projected back to the image planes with the calibrated camera matrix, corrupting by Gaussian noise with a 5 pixel covariance. Based on the initialization from unbiased temporal triangulation, the INT problem can be deduced with a successive coming image until all of the sequential images are processed.

To perform a quantitative evaluation on the proposed propagation-based INT (ININT) method, MVMP, IRMP, NN, and GMRE are applied to perform incremental multiple-view triangulation, respectively. During the INT process, these four compared methods carry out the 3D point estimation repeatedly when a new feature observation arrives, lacking the use of the



**Fig. 2.** Four types of synthetic triangulation instances. (a) Type A: cameras and 3D points are randomly distributed; (b) Type B: the camera moves along a curved trajectory around 3D points; (c) Type C: the camera moves on a circle while 3D points are located at the center; (d) Type D: the camera moves along a curved trajectory towards the 3D scene.

previous triangulated results. In contrast, the proposed INT method considers the consequence of the unbiased temporal triangulation as the seed point, and then only the propagation is executed when encountering a more featured track. It is known that the accuracy of the 3D point is proportional to the number of its observations, and thus the last estimated 3D point with all of the observations is deemed as the most accurate one. Therefore, these 3D points derived from all their observations are projected back to the image planes again to compute the mean 2D reprojection errors. Moreover, the mean 3D errors, the Euclidean distances between generated 3D points with the triangulated ones, are also taken into consideration.

Experimental results are illustrated in Table 1; because iteration is avoided when new observations are added to the INT, the propagation-based INT method is always the fast one. Moreover, both the 3D and 2D errors on different synthetic datasets derived from the proposed INT are superior to the classic MVMP method. Although the seed point of the INT is derived from the unbiased temporal triangulation IRMP, the final 3D results after recursive propagations are close to the continuous IRMP. Compared to the proposed INT method, the GMRE method can obtain slightly better triangulation accuracy, but it takes an enormous amount of runtime. We also find that the accuracy performances of IRMP and NN methods in INT

scenarios are almost the same except the runtime, which is also consistent with our previous convergence analysis.

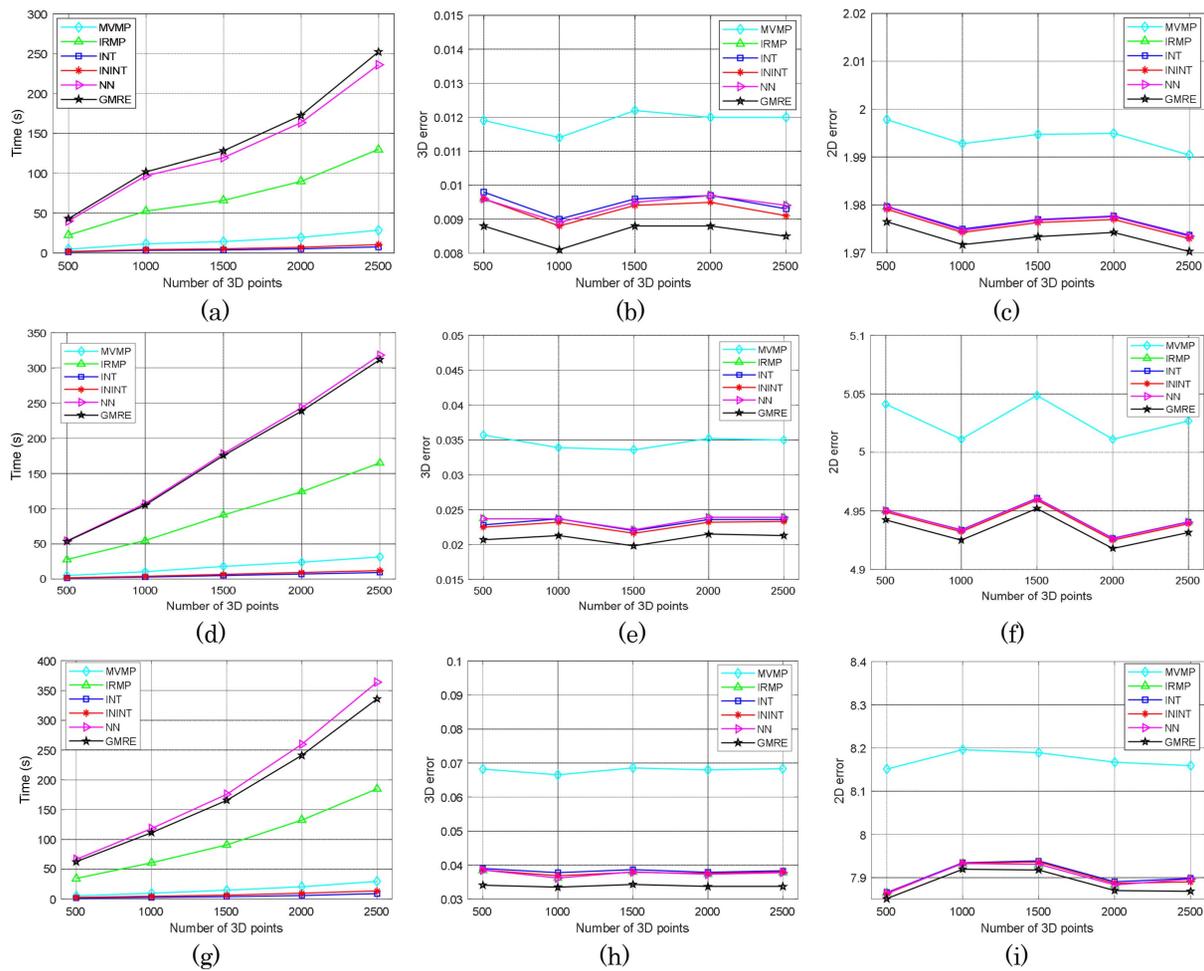
In addition to INT, the ININT method based on a few iterations is applied after the propagated 3D point. Thus, the runtime is a bit longer than the INT method with a superior 3D accuracy. Both the proposed INT and ININT methods can balance the runtime and accuracy in large-scale INTs. The experimental results also demonstrate the convergence ability of the proposed method in different situations.

For further verification of the robustness to varied noises, different Gaussian noise levels are generated to corrupt the 2D observations. Based on the Type D dataset, 2, 5, and 8 pixels covariances are simulated, respectively, while different 3D points are generated, and the performance of different INTs is illustrated in Fig. 3. It is easy to find that the proposed INT (ININT) method is faster than all other methods while obtaining comparable 3D and 2D accuracy, except for the inaccurate MVMP method.

Compared to the previous synthetic dataset Types A, B, and C, the superior performance on the runtime is obvious because the number of feature tracks during INT is large in the Type D dataset. The results also illustrate that the proposed INT (ININT) methods can provide an alternative for large-scale INTs.

**Table 1.** Comparisons between Different Incremental Triangulation Performances.

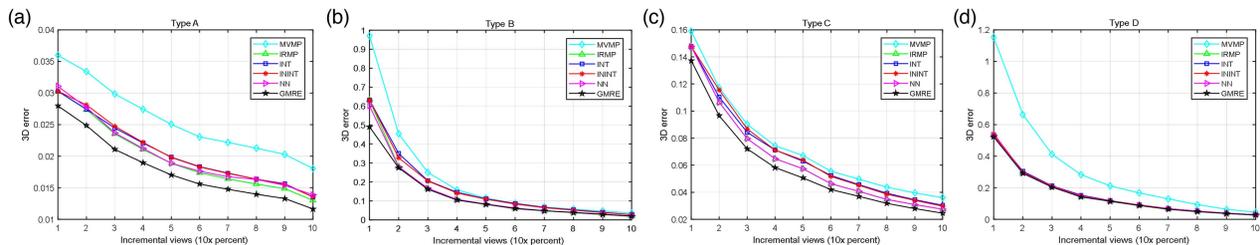
Method	Type A			Type B			Type C		
	Time (s)	3D Error (m)	2D Error (pixels)	Time (s)	3D Error (m)	2D Error (pixels)	Time (s)	3D Error (m)	2D Error (pixels)
MVMP	1.06	0.0192	5.1115	2.45	0.0352	5.0493	2.14	0.0292	4.9511
IRMP	4.01	0.0139	4.7698	11.76	0.0303	4.9635	8.41	0.0251	4.8888
INT	0.68	0.0146	4.7955	0.91	0.0312	4.9649	1.08	0.0258	4.8901
ININT	0.98	0.0145	4.7936	1.45	0.0307	4.9537	1.62	0.0252	4.8895
NN	6.51	0.0139	4.7698	22.60	0.0303	4.9635	14.07	0.0251	4.8888
GMRE	7.08	0.0124	4.7059	22.10	0.0272	4.9621	15.61	0.0237	4.8680



**Fig. 3.** Time and accuracy analysis with different noise levels on Type D synthetic data. (a), (b), and (c) are the results of the time, 3D error, and 2D error when the Gaussian covariance is set as 2 pixels; (d), (e), and (f) are the results of the time, 3D error, and 2D error when the Gaussian covariance is set as 5 pixels; (g), (h), and (i) are the results of the time, 3D error, and 2D error when the Gaussian covariance is set as 8 pixels.

In addition to the efficiency and accuracy evaluation of the proposed method, the convergence performance during INT is verified in this section, where the 3D locations of the synthetic points are assumed as the ground truth, and INT is carried out when new images arrive one by one. Other methods, such as MVMP, IRMP, NN, and GMRE, are also applied to perform the multiple-view triangulation repeatedly after collecting all of the observation vectors.

Due to the length of the feature track for every 3D point being varied during INT, the overall convergence performance of the INT (ININT) method is characterized with the percentage of the views regardless of the number of views, where the total number of the feature track is seen as the 100%, and the mean 3D errors along with INT are calculated with the mean 3D error. Given different synthetic datasets, the results of INT are depicted in Fig. 4. It is easy to find that all the triangulation methods can



**Fig. 4.** Overall convergence of INT with different datasets. (a) Convergence curve of Type A dataset; (b) convergence curve of Type B dataset; (c) convergence curve of Type C dataset; (d) convergence curve of Type D dataset.

Table 2. Runtime Comparisons of Different Methods with Real Datasets<sup>a</sup>.

Data	Total Runtime (s)						Last Mean 3D Error					
	MVMP	IRMP	INT	ININT	NN	GMRE	MVMP	IRMP	INT	ININT	NN	GMRE
E	113.40	941.01	125.16	204.96	1477.41	1386.46	0.0148	0.0131	0.0139	0.0134	0.0131	0.0124
F	110.15	749.16	63.84	110.93	1187.04	1195.77	0.0020	0.0010	0.0014	0.0011	0.0010	0.0002
G	97.99	469.11	83.55	138.63	797.93	665.38	0.0077	0.0072	0.0075	0.0072	0.0072	0.0067
H	23.29	206.40	21.24	35.97	295.81	246.67	0.0018	0.0009	0.0012	0.0010	0.0009	0.0004

<sup>a</sup>E, Lund Cathedral; F, Orebro Castle; G, Ystad Monestary; H, Skansen Kronan.

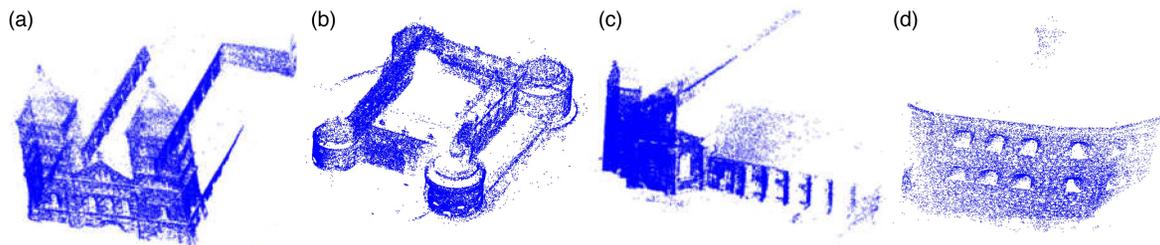


Fig. 5. INT based on real datasets. (a) Lund Cathedral, (b) Orebro Castle, (c) Ystad Monestary, and (d) Skansen Kronan.

achieve a higher 3D accuracy with more increased feature observations, and the achievable 3D error seemingly decays quadratically as more views are added to the system<sup>[13]</sup>. Compared to other methods (MVMP, IRMP, NN, and GMRE), by collecting all of the observations of the 3D points, the proposed propagation-based INT (ININT) with a recursive update about the current observation also achieves similar convergence performance.

#### 4.2. Experiments on real data

We also evaluate our INT method by the publicly available large-scale dataset<sup>[27]</sup>. During the experiments, more views are thus added to the system one by one after initialization, and then the performance of the different methods with the real datasets is illustrated in Table 2 and Fig. 5. The results show that the proposed INT (ININT) is more efficient than the other methods. Even compared to the inaccurate MVMP, the runtime of INT has a comparative advantage. In addition, the 3D errors of the scene points are also calculated to illustrate the accuracy performance of the proposed methods. For convenient evaluation, the 3D point locations provided by the dataset are considered as the ground truth. It is also obvious to find that the proposed INT (ININT) can approach a reliable triangulation performance during INT.

### 5. Conclusion

An efficient INT method based on propagation for sequential multiple views is proposed. Instead of collecting all of the observations of the 3D point to be located, the proposed INT method

only updates the observations of the current view, greatly outperforming the current popular  $L_2$ -norm triangulation method in processing time. Besides, based on the result of INT as an initial value, a novel ININT method with few iterations is derived to obtain a better INT accuracy. Experimental results are presented in detail, which validate that the proposed method is more efficient while achieving comparable accuracy in INT. Nevertheless, as a two-step  $L_2$ -norm-based triangulation method, the accuracy of the proposed method needs reliable initial 3D point estimation. The proposed method can be integrated to accelerate other multiple-view geometry problems, such as the camera pose estimation derived from the PnP, providing high-efficiency 3D reconstruction for robot localization and visual measurements, which is also our future work.

#### Acknowledgement

This work was partially supported by the National Key Research and Development Program of China (No. 2019YFC0119200), the Beijing Natural Science Foundation (No. 3204050), the Open Project Program of State Key Laboratory of Virtual Reality Technology and Systems (No. VRLAB2020B05), and the Fundamental Research Funds for the Central Universities (Nos. 2019RC26 and 2019RC28).

#### References

1. C. T. Seo, S. W. Kang, and M. Cho, "Three-dimensional free view reconstruction in axially distributed image sensing," *Chin. Opt. Lett.* **15**, 081102 (2017).

2. L. Li, Z. Pan, H. Cui, J. Liu, S. Yang, L. Liu, Y. Tian, and W. Wang, "Adaptive window iteration algorithm for enhancing 3D shape recovery from image focus," *Chin. Opt. Lett.* **17**, 061001 (2019).
3. M. Chen, Y. Tang, X. Zou, K. Huang, L. Li, and Y. He, "High-accuracy multi-camera reconstruction enhanced by adaptive point cloud correction algorithm," *Opt. Laser Eng.* **122**, 170 (2019).
4. Z. Zhang, C. Sun, and P. Wang, "Two-step pose estimation method based on five reference points," *Chin. Opt. Lett.* **10**, 071501 (2012).
5. J. Zhuang, C. Hou, Y. Tang, Y. He, Q. Guo, Z. Zhong, and S. Luo, "Computer vision-based localization of picking points for automatic litchi harvesting applications towards natural scenarios," *Biosyst. Eng.* **187**, 1 (2019).
6. Y. Tang, M. Chen, C. Wang, L. Luo, J. Li, G. Lian, and X. Zou, "Recognition and localization methods for vision-based fruit picking robots: a review," *Front. Plant Sci.* **11**, 510 (2020).
7. J. Schonberger and J. Frahm, "Structure-from-motion revisited," in *Proceedings of IEEE International Conference on Computer Vision* (2016), p. 4104.
8. J. H. Kim and B. K. Koo, "Linear stratified approach using full geometric constraints for 3D scene reconstruction and camera calibration," *Opt. Express* **21**, 4456 (2013).
9. R. Mur-Artal, J. Montiel, and J. Tardos, "ORB-SLAM: a versatile and accurate monocular slam system," *IEEE Trans. Robot.* **31**, 1147 (2015).
10. C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: fast semi-direct monocular visual odometry," in *Proceedings of IEEE International Conference on Robotics and Automation* (2014), p. 15.
11. S. Lee and J. Civera, "Closed-form optimal two-view triangulation based on angular errors," in *Proceedings of IEEE International Conference on Computer Vision* (2019), p. 2681.
12. Z. Kukulova, T. Pajdla, and M. Bujnak, "Fast and stable algebraic solution to L2 three-view triangulation," in *Proceedings of the International Conference on 3D Vision* (2013), p. 326.
13. A. Scholefield, A. Ghasemi, and M. Vetterli, "Bound and conquer: improving triangulation by enforcing consistency," *IEEE Trans. Pattern Anal. Mach. Intell.* **42**, 2321 (2020).
14. R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. (Cambridge University, 2004).
15. R. Hartley and P. Sturm, "Triangulation," *Comput. Vis. Image Und.* **68**, 146 (1997).
16. P. Lindstrom, "Triangulation made easy," in *Proceedings of IEEE International Conference on Computer Vision* (2010), p. 1554.
17. H. Stewenius, F. Schaffalitzky, and D. Nister, "How hard is 3-view triangulation really?" in *Proceedings of IEEE International Conference on Computer Vision* (2005), p. 686.
18. R. Hartley and F. Schaffalitzky, "L<sub>∞</sub> minimization in geometric reconstruction problems," in *Proceedings of IEEE Computer Vision and Pattern Recognition* (2004), p. 504.
19. H. Li, "A practical algorithm for L<sub>∞</sub> triangulation with outliers," in *Proceedings of IEEE Computer Vision and Pattern Recognition* (2007), p. 1.
20. G. Zhou, Q. Wang, and Z. Xiao, "Robust outlier removal using penalized linear regression in multiview geometry," *Neurocomputing* **267**, 455 (2017).
21. Q. Zhang and T. Chin, "Coresets for triangulation," *IEEE Trans. Pattern Anal. Mach. Intell.* **40**, 2095 (2018).
22. S. Recker, M. Hess-Flores, and K. Joy, "Fury of the swarm: efficient and very accurate triangulation for multi-view scene reconstruction," in *Proceedings of IEEE International Conference on Computer Vision Workshops* (2013), p. 652.
23. S. Lee and J. Civera, "Triangulation: why optimize?" in *Proceedings of the British Machine Vision Conference* (2019), p. 1.
24. K. Yang, W. Fang, Y. Zhao, and N. Deng, "Iteratively reweighted midpoint method for fast multiple view triangulation," *IEEE Robot. Autom. Lett.* **4**, 708 (2019).
25. B. Micusik and T. Pajdla, "Structure from motion with wide circular field of view cameras," *IEEE Trans. Pattern Anal. Mach. Intell.* **28**, 1135 (2006).
26. V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: an accurate(n) solution to the pnp problem," *Int. J. Comput. Vision* **81**, 155 (2009).
27. O. Enqvist, F. Kahl, and C. Olsson, "Non-sequential structure from motion," in *Proceedings of IEEE International Conference on Computer Vision Workshops* (2011), p. 264.