Optics and Laser Technology 110 (2019) 165–175

Contents lists available at ScienceDirect

Optics and Laser Technology

journal homepage: www.elsevier.com/locate/optlastec

Full length article

GPU-accelerated feature tracking for 3D reconstruction

Mingwei Cao[a,b], Wei Jia[a,b], Shujie Li[a,b,*], Yujie Li[c], Liping Zheng[a,b], Xiaoping Liu[a,b]

[a] School of Computer Science and Information Engineering, Hefei University of Technology, Hefei, China
[b] Anhui Province Key Laboratory of Industry Safety and Emergency Technology, Hefei, China
[c] Fukuoka University, Fukuoka, Japan

HIGHLIGHTS

- The proposed method has fast speed.
- The proposed method can efficiently remove outliers.
- The proposed method is very suitable to 3D reconstruction with UAVs.
- The proposed method can help SFM system to produce high-quality 3D model.

ARTICLE INFO

Keywords:
Feature tracking
3D reconstruction
Local feature
GPU
Structure from motion

ABSTRACT

3D reconstruction based on structure from motion is one of the most techniques to produce sparse point-cloud model and camera parameter. However, this technique heavily relies on feature tracking method to obtain feature correspondences, then resulting in a heavy computation burden. To speed up 3D reconstruction, in this paper, we design a novel GPU-accelerated feature tracking (GFT) method for large-scale structure from motion (SFM)-based 3D reconstruction. The proposed GFT method consists of GPU-based Gaussian of image (DOG) keypoint detector, RootSIFT descriptor extractor, k nearest matching, and outlier removing. Firstly, our GPU-based DOG implementation can detect thousands of keypoints in real-time, whose speed is 30 times faster than that of the CPU version. Secondly, our GPU-based RootSIFT descriptor can compute thousands of descriptors in real-time. Thirdly, our GPU-based descriptor matching is 10 times faster than that of the state-of-the-art methods. Finally, we conduct thorough experiments on different datasets to evaluate the proposed method. Experimental results demonstrate the effectiveness and efficiency of the proposed method.

1. Introduction

3D reconstruction is an important topic in the fields of computer vision and graphics due to its potential applications, such as virtually reality [1], augmented reality [2], city-scale modeling [3], visualization [4], image-based localization [5], cross-modal retrieval [6,7], pose estimation [8], change detection [9], object tracking [10,11], navigation [12] and autonomous driving [13]. Thus, many 3D reconstruction techniques have been proposed for various tasks. Among them, structure from motion (SFM) is one of the most famous techniques and has been received wide attentions from the academic world and the industrial world. Generally, SFM is a collection of techniques including feature tracking [14], camera calibration [15], pose estimation, motion averaging [16], perspective-n-point (PnP) [17], registration [18], triangulation [19] and bundle adjustment [20]. The SFM system can produce sparse point clouds and camera parameters from the given image collection.

With the increase of amount of image dataset, nowadays, the SFM systems have been obtained significantly progress. For Instance, some state-of-the-art SFM systems including incremental SFM architecture and global SFM architecture can produce accurate 3D point clouds for large-scale scenes [21,22]. However, these SFM systems are very time-consuming. According to the newest survey made by Ozyesil et al. [23], the quality of point cloud produced by COLMAP [22] is rank 1 among the existing SFM systems, which can not only produce high-precision camera parameters, but also can produce high-quality 3D model. But, the COLMAP system has high computation time for large-scale outdoors, even it was accelerated with parallel computing techniques. Thus, various strategies have been made to reduce the computation burden of the SFM systems. For example, Wu et al. [24] hold that bundle adjustment is time consuming for large-scale scenes, thus, a multicore bundle adjustment method has been proposed to optimize the

* Corresponding author at: School of Computer Science and Information Engineering, Hefei University of Technology, Hefei, China.
E-mail address: lisjhfut@hfut.edu.cn (S. Li).

https://doi.org/10.1016/j.optlastec.2018.08.045
Received 30 June 2018; Received in revised form 11 August 2018; Accepted 23 August 2018
Available online 20 September 2018
0030-3992/

initial point clouds to obtain compact point-clouds. David et al. [25] proposed a discrete continuous optimization method for large-scale structure from motion, in which the initial optimization step is done by a discrete Markov random filed (MRF), and implemented in parallel architecture. After a series of matrix factorization operations, this system can produce point-clouds for the scene. Crandall et al. [26] proposed large-scale SFM based on graph partitioning, in which a divide and conquer mthod is used to partition the image data set into smaller sets or components which are reconstructed independently. Once the model of each independent sets is reconstructed, the model of the whole scene can be obtained by combing them. Sweeney et al. [27] proposed a distributed camera model for large-scale SFM, in which the incrementally adding one camera at a time to grow the reconstruction is replaced by a distributed camera, then the camera parameters can estimated in simultaneous, this can significantly reduce computational cost.

Although some useful methods have been proposed to improve the efficiency of 3D reconstruction based on SFM, however, 3D reconstruction is still time consuming according to the report of Saputra et al. [28]. Particularly, the most time-consuming step in SFM is feature tracking. To speed up 3D reconstruction, Zhang et al. [29] proposed an effective non-consecutive feature tracking (ENFT) method for SFM-based 3D reconstruction, the ENFT relies heavily on two-pass matching to improve the precision of descriptor matching. However, the two-pass matching has also high computational time under large-scale scenes with repeating features. With the development of graphics process units (GPUs), some time-consuming methods cloud be accelerated using parallelization technique. For example, Sudipta et al. [30] implemented KLT-GPU method with CUDA to improve the efficiency of original KLT. Garcia et al. [31] implemented GPU-based k-nearest neighbor search (KNN) to match high-dimensional feature descriptor.

Inspired by the thought of GPU-acceleration, in this paper, we propose GPU-accelerated feature tracking (GFT) method for 3D reconstruction based on SFM. In the proposed method, we firstly parallelize the Difference of Gaussian (DOG) operation [32] to accelerate keypoint detection; Secondly, the RootSIFT descriptor extractor is parallelized to get robust description for the selected keypoints; Thirdly, the k nearest neighbor (KNN) method is parallelized to match descriptors; Finally, the vector field [33] based method is utilized to remove outliers from the initial matches, which will result in a set of correct matches. Our work has a broad of interests to the 3D reconstruction, computer vision and computer graphics community since many of the key steps in the proposed method are shared by other methods, which can also be accelerated on the GPU.

The main contributions of this work are summarized as follows:

- A GPU-accelerated feature tracking method is proposed for large-scale SFM, which significantly improve the efficiency of 3D reconstruction, in which the DOG keypoint detector and RootSIFT descriptor extractor have been parallelized. As a result, the efficiency of 3D reconstruction system can be significantly improved.
- A novel mismatch removing algorithm based on vector field is proposed to remove outliers from initial matches, which can efficiently avoid the ambiguation of point clouds produced by the SFM system.
- We discuss various factors which may affect the performance of feature tracking method, this can be as a guide to design an excellent feature tracking method for large-scale 3D reconstruction based on SFM.

The rest of this paper is organized as follows: the related work is presented in Section 2. The proposed GFT method is presented in Section 3. In Section 4, a comparative experiment is presented to evaluate the proposed method. The conclusion and remarking comments are presented in Section 5.

2. Related work

In this section we will briefly review some existing works including feature tracking and 3D reconstruction methods based on SFM technique to better understand the proposed GFT method.

2.1. Feature tracking

In the past decade, many feature tracking methods have been proposed [30,31,34–45] for 3D reconstruction. One of the most famous is Kanade Lucas Tomasi (KLT) [34–36] method, which uses optical flow to track keypoints appeared in the next frame of video. To speed up KLT, Sinha et al. [30,43] implemented KLT on graphics processing unit (GPU), which is named as KLTGPU. However, the KLT-like methods are easily to produce incorrect matches, these errors are unacceptable for feature tracking in SFM and SLAM [40]. To improve the precision of KLT-like methods, Myung et al [46] proposed to use inertial measurement units (IMU) to assist KLT for reducing error accumulation. However, the KLT-like methods easily suffer from illumination and scale changes, which can aggravate feature drifting in SFM [47,48] and SLAM [49,50] under the outdoors.

Recently, feature tracking based on feature detection and matching framework (DMF) have received wide attentions from the communities of compute vision [37,41] and computer graphics [51]. For example, Zhang et al. [37] proposed an efficient feature tracking algorithm for non-consecutive frames for SFM, in which they use two-pass matching to process the occlusions to avoid feature drifting. In order to improve the robustness of feature tracking for simultaneous localization and mapping (SLAM), Garrigues et al. [52] proposed a semi-dense point tracking algorithm to produce dense trajectories for the mobile devices. Lee et al. [39] proposed a hybrid feature tracking using optical flow to detect distinctive invariant feature points for marker-less augmented reality. In Bundler system, SIFT and BF method is used to detect and match keypoints, resulting in a high computation time. Buchanan et al. [53] proposed an interactive feature tracking (IFT) method, which use KD-Tree and dynamic programming techniques to speed up feature descriptors matching. Another purpose pursued by feature tracking method is to obtain accurate feature correspondences. To achieve this purpose, Zhang et al. [54] proposed to use epipolar geometry constraint to remove outliers neared its epipolar line. Wu et al. [55] proposed to use viewpoint-invariant patches (VIP) to match images with high resolution, then result in a collection of accurate matches. However, the VIP feature is time consuming, thus, this can decrease the efficiency of feature tracking. To improve time efficiency, the RANSAC procedure is used to remove outliers, such as [56,57]. Lee et al. [39] proposed hybrid feature tracking (HFT) method for augmented reality, which use multiple strategies including optical flow, RANSAC, epipolar constraint to remove outliers, then result in a collection of correct matches. Moreover, using a fast feature detector can accelerate the process of feature tracking, for example, Zach et al. [58] propose to use SUFR feature to replace SIFT to obtain a fast speed. Svarm et al. [59] proposed a graph-theoretical approach to point tracking, and used Gomory-Hu algorithm [60] to remove incorrect matches.

The most recently, feature tracking methods for mobile devices have attracted wide attentions from computer vision community, one of the most famous method is proposed by Garrigues et al. [61], which can efficiently produce accurate and dense feature point trajectories in real-time. With development of subspace learning theory, it provides a new approach for research of feature tracking. For example, Poling et al. [40] proposed a better feature tracking method through subspace constraints (BFT) for jointly tracking a set of features, which enables sharing information between the different features in the scene. The experimental results made in [40] show that the proposed method can utilized to track keypoints for both rigid and non-rigid objects. Jia et al. [45] proposed a novel framework based on low-rank structures, termed ROML, for feature tracking. ROML optimizes simultaneously a partial

permutation matrix (PPM) for each image, and feature correspondences are established by the obtained PPMs. Zhao et al. [62] hold that most existing feature tracking methods are incapable of effectively modeling and balancing the following three aspects in a simultaneous manner: temporal model coherence across frames, spatial model consistency within frames, and discriminative feature construction. To address this problem, they proposed a robust feature tracking method based on spatio-temporal multi-task structured output optimization driven by discriminative metric learning.

Although, existing feature tracking methods as mentioned before try to use the different strategies to improve the stability and efficiency, respectively. Obviously, it would be better if the efficiency and stability of feature tracking are considered at the same time, especially in large-scale SFM system.

2.2. 3D reconstruction

In the past years, many 3D multi-view 3D reconstruction systems based on SFM technique have been proposed. The existing SFM systems can be roughly divided into two categories, namely incremental SFM and global SFM.

For the former, Bundler is the most famous SFM system, which is developed by Snavely et al. [63]. Bundler can reconstruct spare point-cloud model and camera parameters from unordered image collections, which consists of camera calibration, feature tracking, camera pose estimation including relative pose and absolute pose, triangulation, and bundle adjustment. In the Bundler system, the authors employ scale invariant feature transform (SIFT) [32] to detect keypoints and compute descriptors, and use brute-force matching (BFM) strategy to match descriptors for image pairs. However, owing to the usage of SIFT and BFM, the Bundler system has high computation burden especially in large-scale 3D reconstruction with several thousands of images. This problem has been pointed by the work of Agarwal et al. [64]. To save the computation time for 3D reconstruction based SFM, Zach et al. [65] exploited exploits speeded up robust features (SURF) to detect keypoint and compute feature descriptor for feature tracking, and developed a fast SFM system named ETH-3D. Many previous works have proved that the speed of ETH-3D system is fast than that of Bundler [66]. Dong et al. [67] proposed a robust markerless real-time camera tracking system based on keyframe selection and recognition, named ACTS, for multi-view 3D reconstruction. The ACTS system includes an offline module to select features from a group of reference images and an online module to match them to the input live vide in order to quickly estimate the camera pose. Moreover, to accelerate the speed of feature tracking in ACTS, the authors use parallelized-SIFT named SFITGPU [68], to detect keypoints and compute descriptors for the selected keypoints. Later, ACTS was extended to LS-ACTS for large-scale outdoor environments' applications [54]. Based on Bundler, Wu et al. [69] developed a Visual SFM (VSFM) system, which also uses SIFTGPU to detect keypoint and compute descriptor in feature tracking for saving computational cost. In addition to the promising speed, the VSFM system has an excellent graphic user interface (GUI) to make operation easily, and can work with the patch-based multi-view stereo system (PMVS) [70] to produce dense 3D geometry of the scene. Ni et al. [71] proposed a novel algorithm that solves the SFM problem in a divide and conquer manner by exploiting its bipartite graph structure. Thus, the proposed HyperSFM system use a hypergraph representation to recursively divide an SFM system, in which finding edge separators yields the desired nonlinear sub-problems.

For the latter, Moulon et al. [72] proposed a novel global calibration approach based on the global fusion of relative motions between image pairs for robust, accurate and scalable SFM. After an efficient contrario trifocal tensor estimation, the authors define an efficient translation registration method to recover accurate positions. Besides accurate camera position, Moulon et al. use KAZE [73] feature to detect keypoints in feature tracking process, then resulting in a high-precision matching score, this can significantly improve the quality of 3D model. Based on optimized viewgraph, Chris et al. [74] designed and implemented an excellent SFM system, named Theia-SFM, to produce compact and accurate point-cloud model for both indoor and outdoor scenes. Based on the successes in solving for global camera rotations using averaging technique, Kyle et al. [75] proposed a simple, effective method for solving SFM problems by averaging epipolar geometries. This method has two main insights. First. A simple method is proposed for removing outliers from feature tracking by solving simpler low-dimensional sub-problems named 1DSFM. Second, these authors present a simple, principled averaging technique to improve the robustness of 3D reconstruction system. With the development of depth-camera such as Kinect, ASUS Xtion Pro, Intel RealSense and Matterport Pro, RGB-D datasets are easily to capture, and are widely used in 3D reconstruction. As a result, to effective use RGB-D datasets, Xiao et al. [76] developed RGBD-SFM system to construct 3D geometry, in which owing to the prior information, depth-map, is used, the quality of 3D model is significantly improved. Sid et al. [77] held that the semantic information can help to reconstruct complete 3D model. Based on this important discovery, they proposed a semantic SFM (SSFM) system. To deal with the moving objects in scene, Wang et al. [78] designed a dynamic SFM system, which can detect scene changes from image pairs.

Up to now, most of SFM systems are require the points in scene must be appeared in three views at least. In order to defend this drawback, Zheng et al. [79] proposed to use structure-less resection for SFM. In order to speed up the process of multi-view 3D reconstruction, Crandall et al. [80] proposed to use discrete-continuous optimization for large-scale SFM and implemented this system in parallel pipeline.

3. The proposed method

The pipeline of the proposed method is depicted in Fig. 1. For a given image pairs, firstly, the DOG-GPU keypoint detector is used to detect keypoints; Secondly, the RootSIFT descriptor extractor is used to compute descriptor; Thirdly, the KNN is used to find matches for the two descriptor sets; Finally, to remove mismatches, the VFC algorithm is used, then the correct matches can be obtained. In the following sections, we will describe how to implement the DOG keypoint detector, RootSIFT descriptor extractor, and KNN on GPU device with CUDA kernel.

3.1. Feature detection

The DOG detector is firstly introduce in [32], which is wide use in various computer vision tasks. To locate a DOG keypoint for image I, as shown in Fig. 2, the following steps are required: construct multiple-scale spaces, keypoint localization, orientation assignment. One of the most computational cost steps in DOG is to construct multiple-scale spaces, in which too many convolutional operations should be conducted.

Specifically, for a given image, $I(x, y)$, the scale-space is defined as a function, $S(x, y, \sigma)$, which is produced from the convolution of a variable-scale Gaussian, $G(x, y, \sigma)$:

$$S(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{1}$$

where $*$ represents the convolution operation in x and y, and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-x^2+y^2/2\pi^2} \tag{2}$$

Then, the difference-of-Gaussian image, $D(x, y, \sigma)$, which can be computed by the difference of two nearly scales:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) = S(x, y, k\sigma) - S(x, y, \sigma) \tag{3}$$

Thus, the multiple-scale spaces, $MSS(x, y, \sigma)$, that is computed by changing the values of k and σ:

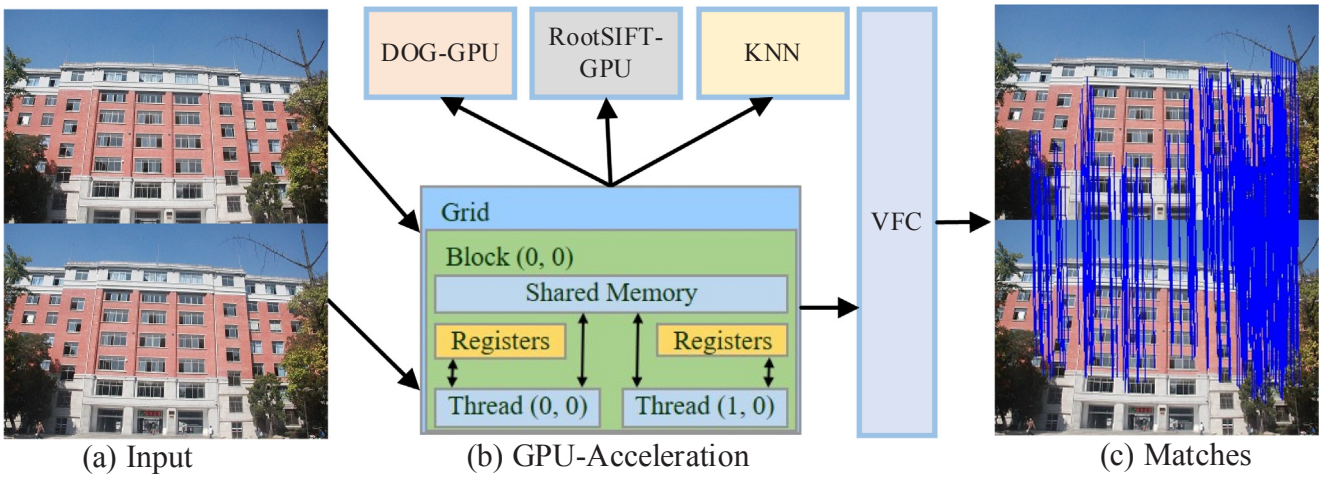

Fig. 1. The pipeline of GPU-accelerated feature tracking.

$$MSS(x, y, \sigma) = Stack_{k_i}^{\sigma_i}(S(x, y, k_i\sigma_i) - S(x, y, \sigma_i)) \tag{4}$$

where $Stack_{k_i}^{\sigma_i}(*)$ is the vertical connection of difference of Gaussian image.

Therefore, parallelized convolutional operation is the most important step to accelerate DOG detector. As shown in Fig. 3, the convolutional operations are conducted in the multiple Blocks, which is a GPU execution unit. We set the number of Block be 10, and each block contain 20 threads. These can be changed according to the usage of GPU device to achieve the best performance.

3.2. Descriptor extraction

It is well known that SIFT descriptor has more robustness than that of SURF, BRIEF and ORB due to its main direction and histogram representation. Recently, RootSIFT [81] descriptor has shown better robustness than that of original SIFT descriptor. Thus, to improve the quality of 3D models, the RootSIFT descriptor extractor is used to produce a robust description for the selected DOG-GPU keypoint. Once a DOG-GPU keypoint has been selected, as shown in Fig. 4, the feature descriptor is computed as a set of orientation histograms on 4×4 pixel neighborhoods. The orientation histograms are relative to the orientation DOG-GPU keypoint, the orientation data comes from the Gaussian image closet in scale to the keypoint's scale.

For a given keypoint, $k(x, y)$, with respect to the SIFT descriptor, $D_s(d_1, \cdots, d_{128})$, then the RootSIFT descriptor, $D_{rs}(d_1, \cdots, d_{128})$, can be computed by the following formula:

$$D_{rs}(i) = \sqrt{\frac{D_s(i)}{\sum_{j=1}^{128} D_s(j)}} \tag{5}$$

Once, the RootSIFT descriptors have been computed, we can use Hellinger distance [81] to measure the similarity of the two descriptors.

It should be noted that the number of Block is set to 20, and each Block contain 128 threads. Each thread may compute one element of the descriptor. Thus, the number of threads in Block is assigned to equal to that of descriptor length. Once the keypoint is detected, then the descriptor may be computed simultaneously by 128 threads. As a result, from keypoint detection to descriptor computing, the process has little latency.

3.3. Feature matching

The purpose of descriptor matching is to measure the similarity of two descriptors. In SURF and SIFT, the L2 distance is used to measure the similarity of descriptors because the type of descriptors is float number [32]. For binary descriptors, such as BRIEF and LDB, hamming distance is used [82]. Here, we use a Hellinger kernel instead the L2 distance to measure the similarity between RootSIFT descriptors, results in a significant performance boost in process of feature tracking.

For two L1 normalized histogram, v_x and v_y, the Hellinger kernel [81] can be defined as:

$$\mathrm{H}(x, y) = \sum_{i=1}^{n} \sqrt{v_x^i v_y^i} \tag{6}$$

Where, $\sum_{i=1}^{n} v_x^i = 1$ and $v_x^i \geq 0$.

Thus, the similarity of two RootSIFT descriptors can be calculated by the following formula:

$$D_s(v_x, v_y)^2 = 2(1 - H(v_x, v_y)) \tag{7}$$

Specifically, we search for the two nearest neighboring features of k_t in I_{t+1} with respect to the Hellinger distance of the descriptor vectors and denote then as $N_{t+1}^1(k_t)$ and$N_{t+1}^2(k_t)$. Their corresponding descriptor vectors are denoted as $\mathrm{D}(N_{t+1}^1(k_t))$ and $\mathrm{D}(N_{t+1}^2(k_t))$ respectively. The matching confidence between k_t and $N_{t+1}^1(k_t)$ is defined as:

$$\mathrm{c} = \frac{D_s(k_t, N_{t+1}^1(k_t))^2}{D_s(k_t, N_{t+1}^2(k_t))^2} \tag{8}$$

If $\mathrm{c} < \varphi$, we assign $k_{t+1} = N_{t+1}^1(k_t)$ and mark these detected keypoints as matched features. In our experiments, φ is set to 0.8.

3.4. Remove outliers

Matches by descriptors matching are often include some incorrect matches, also called outliers, which can produce disambiguate point clouds when it used in triangulation. To remove these outliers from matches, various strategies have been proposed, such as RANSAC-based method [83], statistics-based method [84], and ratio test [32]. However, these methods have intrinsic deficiency. For example, RANSAC-

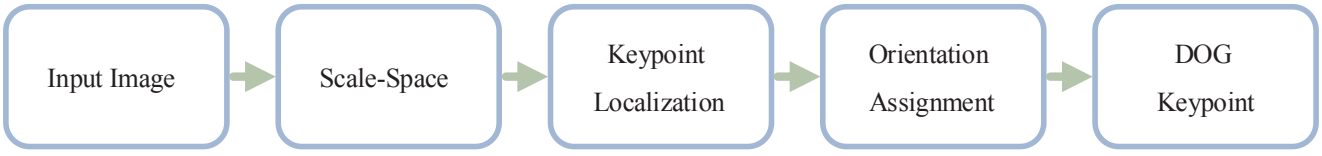

Fig. 2. The pipeline of DOG keypoint localization.

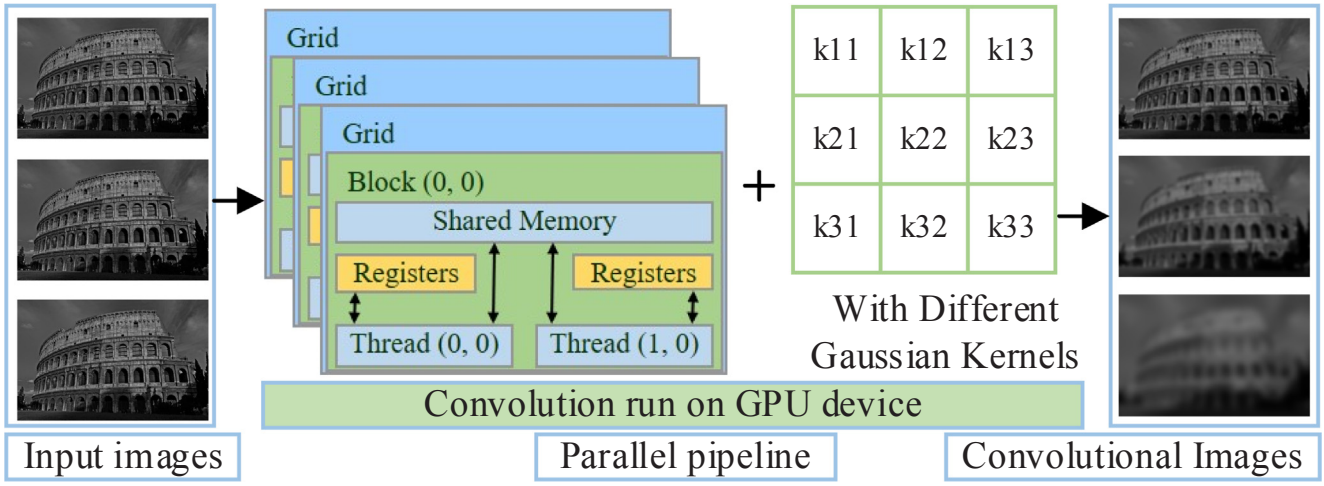

Fig. 3. Parallel image convolution.

based methods require the precondition that the number of inliers must be greater than fifty percent. The statistics-based method heavily relies on the matching results of its neighbor. The performance of ratio test largely depends on the threshold.

Recently, the vector filed consensus-based method (VFC) [33] has obtained an excellent performance, so we use VFC to remove mismatches. For a given set of observed input-output pairs $s = \{(x_n, y_n) \in X \times Y\}_{n=1}^{N}$, our purpose is to lean a mapping f: X → Y to fit the inliers well. Thus, the likelihood can be defined as:

$$\rho(\mathrm{Y}|\mathrm{X}, \theta) = \prod_{n=1}^{N} \rho(y_n|x_n, \theta) \tag{9}$$

Where $\theta = \{f, \gamma^2, \tau^2\}$ is the set of unknown parameters. In this paper, we use EM method to estimate the value of these parameters.

According to the vector-valued theory [33], the optimal function f can be defined as:

$$f(x) = \sum_{n=1}^{N} \Phi(x, x_n) c_n \tag{10}$$

where c_n is the coefficient.

The energy function of the VFC is defined as:

$$\phi(f) = \frac{1}{2\sigma^2} \sum_{n=1}^{N} p_n \, ||y_n - f(x_n)||^2 + \frac{\lambda}{2} ||f||_H^2 \tag{11}$$

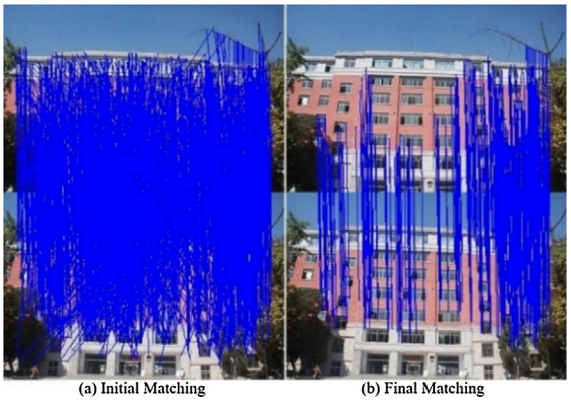

Fig. 5. Remove outliers using VFC-based method, the left figure is the result of the BFM, the right figure is the result of VFC-based method.

Optimizing the formula (3) until convergence, we can obtain the vector filed f and inlier matching set $\mathrm{M} = \{n|p_n > \tau, n = 1, \cdots, N\}$.

Fig. 5 presents the initial matching and final matching, in which the former is generated by BFM method, the latter is generated by using VFC method. We see clearly that the VFC method can remove outliers very well.

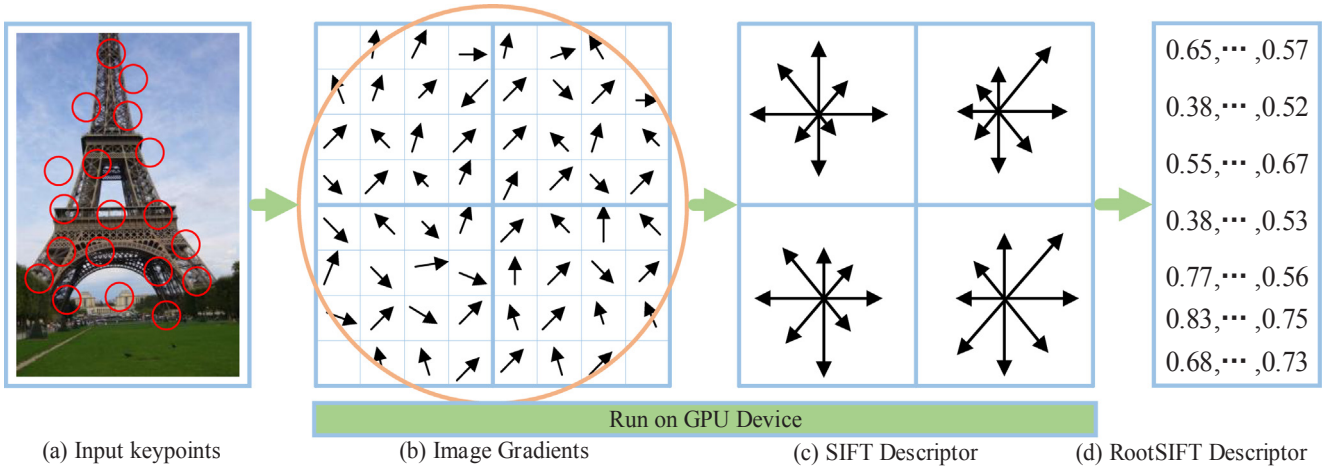

Fig. 4. RootSIFT descriptor.

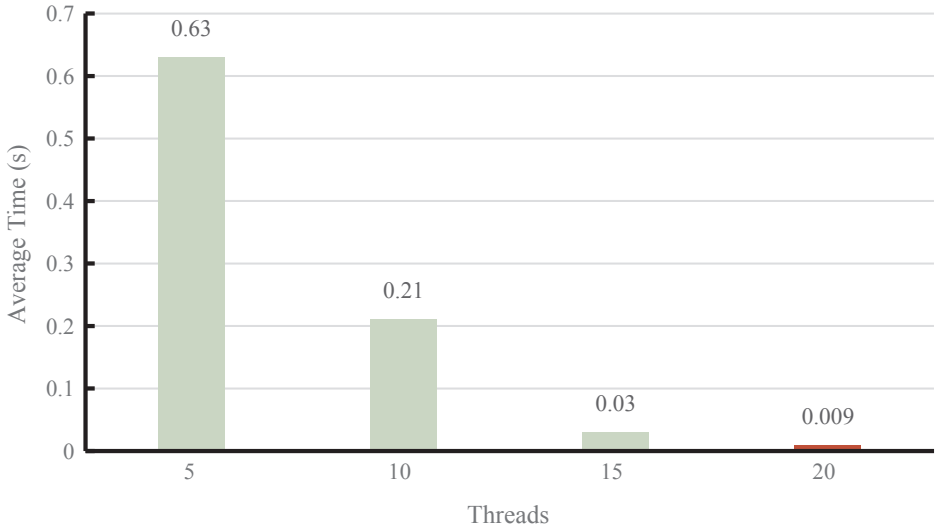

Fig. 6. The latency with variable threads.

4. Experimental results

The proposed GFT method is developed in C++, Nvidia CUDA SDK 8.0 and OpenCV SDK 3.3. To assess the performance of the proposed method, we have evaluated it on the Family dataset [85] and UAV dataset, and make a comprehensive comparison with brute-force matching (BFM) [63], ENFT [29], and MODS [86]. The ENFT is GPU-based method, and implemented in OpenGL and CUDA. The others are CPU-based methods.

4.1. How to set the number of threads

As we well known that the number of threads has impact significantly on the computational cost. However, the number of threads is not unchangeable, which depends on the usage of GPU device and the desired performance. In this section, we will discuss how threads is scheduled to achieve the best performance according to the test on the Oxford dataset [87]. We firstly assign 128 threads to compute descriptor for the selected keypoint because the RootSIFT has 128 bits. In SFM-based 3D reconstruction, each image should have at least 10 keypoints that can be used to produce point clouds. According the background of 3D reconstruction, the number of Blocks in feature detection and descriptor computing is all set to be 10. Up to now, we only need to determine the value of on parameter, namely thread used in feature detection. According to the test on the Oxford dataset, averaging computational cost of descriptor computing is 0.73 s. Thus, to achieve the best performance, the process of descriptor computing must be finished in 0.73 s when the keypoint is located. Once the computation time of descriptor computing exceeds 0.73 s, the parallel system may have higher latency. To this end, we can determine the number of threads in feature detection by the minimum delay theory.

Fig. 6 shows the latency of parallel system with different number of threads, the system has the minimum latency when 20 threads is used in the process of feature detection. As a result, we set the number of threads in feature detection be 20 to achieve the best performance.

4.2. Evaluation on Family dataset

The Family dataset is constructed by Knapitsch et al [85], which is the newest benchmark dataset for performance evaluation of 3D reconstruction based on SFM. The matches of each feature tracking method are depicted in Fig. 7, in which the BFM has the minimum number of matches. The number of matches of the ENFT is greater than that of BFM; The matches of the MODS are greater than that of both BFM and ENFT. The GFT method has the maximum number of matches. In the process of experiment, we found that BFM the relies heavily on the value of threshold which is used to decide the correct matches.

To assess the speed of each feature tracking method, we record the computation time for these methods based on the testing on the Family dataset. Fig. 8 shows the computation time of each method, in which the BFM has the highest computation time, 22.3 s; The proposed GFT has the fastest speed, and is 20 times faster than that of BFM. As a result, according to the experiment, we can find that the GFT has the best performance on both matching confidence and speed.

Fig. 9 presents the point-cloud mode for the Family dataset, generally, this model is very dense. Thus, the GFT method is effective in practice.

4.3. Evaluation on UAV dataset

The UAV dataset is constructed by PIX4D company, which is publicly available dataset[1] for performance evaluation of 3D reconstruction based on SFM. The matches of each feature tracking method are depicted in Fig. 10 where the BFM has the minimum number of matches. The number of matches of the ENFT is greater than that of BFM; The matches of the MODS are the second place among these feature tracking methods. The GFT method has the maximum number of matches. In the process of experiment, we found that the ENFT method may produce many incorrect matches when the image is rotated with more than 180 degrees. However, the GFT method has always produced a plenty of correct matches even the image is rotated with more than 200 degrees.

We record the computation time for these methods based on the testing on the Family dataset for fairly assessing the speed of each feature tracking method. Fig. 11 shows the computation time for each method, in which the BFM has the highest computation time, 31.3 s; The GFT has the lowest computational cost, 1.8 s, which is 17 times faster than that of BFM method.

We also reconstruct the point-cloud model for the UAV dataset, which is depicted in Fig. 12, it can be seen that the shape of the produced model is very clear. Although, some small holes are appeared in the reconstructed model, the point-cloud model is dense in the rich texture areas.

[1] https://support.pix4d.com/hc/en-us/sections/200591139-Example-Datasets-Available-for-Download

Fig. 7. Matches on the Family dataset. The first line is samples; the two last rows are matching results for each method.

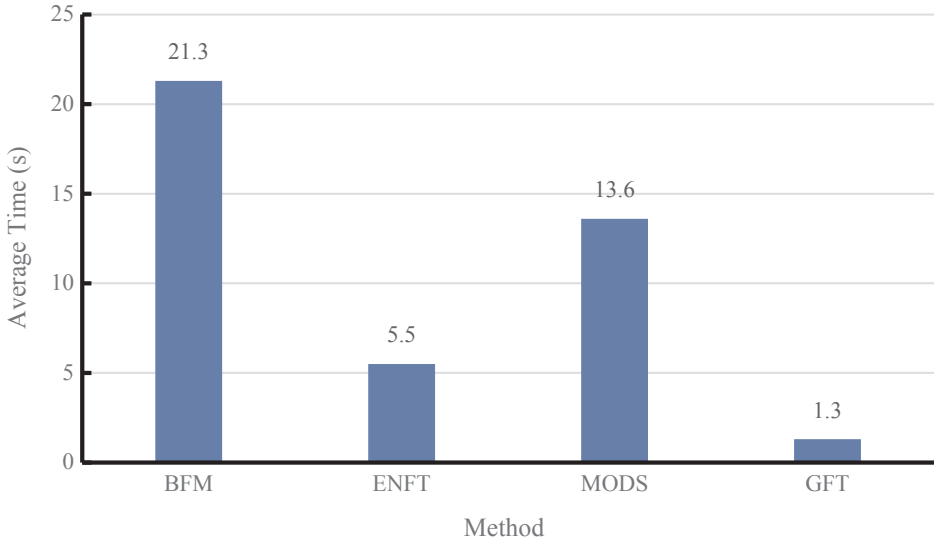

Fig. 8. Computation time of each method on the Family dataset.

4.4. Discussion

As we known that speed and quality are two pursuits of goals from the researchers of feature tracking method. Thus, in this section, we will discuss what impact on the speed and accuracy of feature tracking.

4.4.1. Speed

According to our experiments, the reasons that effects the speed of feature tracking can be summarized as follows:

(1) The binary type descriptors have faster speed than that of float type descriptors. For example, the speed of ORB descriptor has faster speed than that of original SIFT descriptor implemented in [32], the

Fig. 9. The sparse point-clouds of the Family dataset.

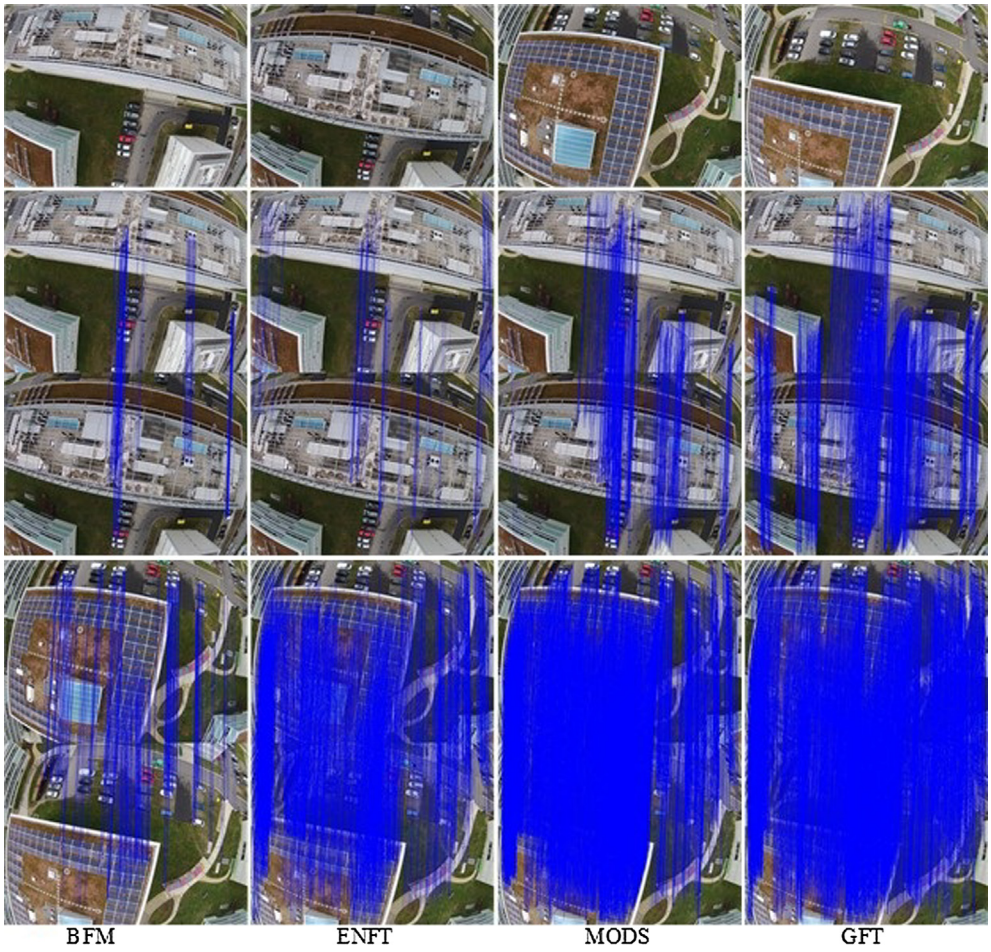

Fig. 10. Matches on the UAV dataset.

former is binary type, and the latter is float type.

(2) The longer of descriptor length, the speed is lower. SIFT descriptor has 128 dimensions, the SURF descriptor has only 64 elements, thus the speed of the latter is always faster than that of the former.

(3) The number of keypoints has also great influence on the speed of feature tracking method. For instance, the ENFT [29] method has variant speed when different local feature is used.

4.4.2. Accuracy

Similarly, the reasons that effects the accuracy of feature tracking method can also be summarized as follows:

(1) The feature tracking method has high accuracy when the float type descriptors are used.

(2) The longer of descriptor length, the feature tracking method has high accuracy. According to our experiments, the ENFT with SIFT feature has higher accuracy than it with SURF feature.

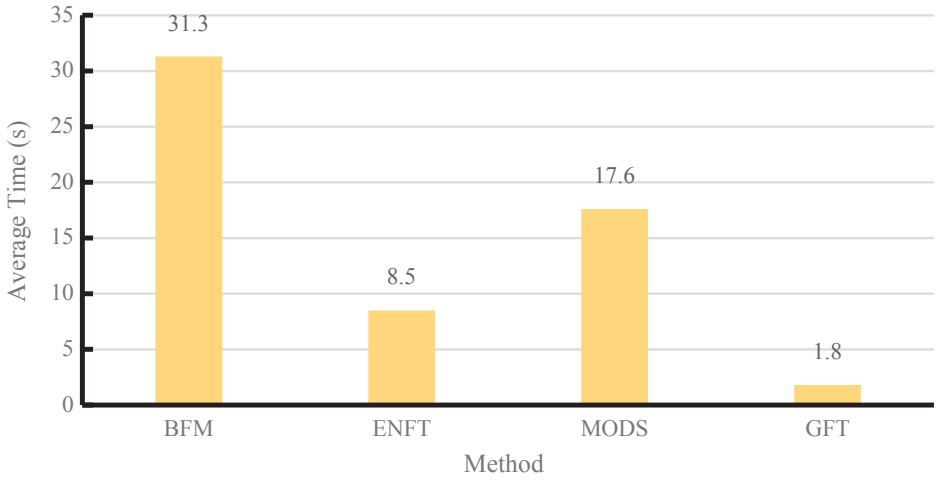

Fig. 11. Computation time of each method on the UAV dataset.

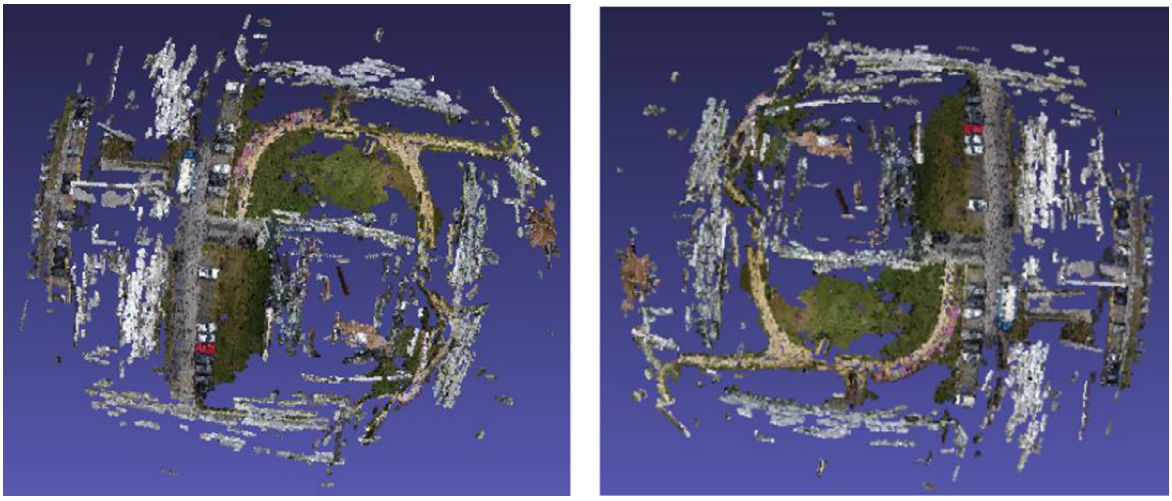

Fig. 12. The sparse point-clouds of the UAV dataset.

(3) The matching strategy has also an important influence on the accuracy of feature tracking method. For example, in RootSIFT descriptor, the Hellinger distance is used to instead of L2 distance to measure the similarity of two descriptors, then resulting in high matching confidence.

5. Conclusion

In this paper, we proposed a GPU-accelerated feature tracking (GFT) method for large-scale 3D reconstruction based on SFM. The proposed GFT method consists of keypoint detection, descriptor computing, descriptor matching and outliers removing. To deal with high-resolution images, we use GPU-accelerated DOG detector to detect keypoint to relieve computation burden. To get robust description for the selected keypoints, the GPU-accelerated RootSIFT descriptor extractor is used, which can not only speed up descriptor extraction, but also can produce a robust description. Moreover, the vector filed-based procedure is used to remove incorrect matches, this could efficiently avoid the ambiguation of 3D model. The proposed method is versatile and expansible, which can be easily extended to other applications where feature tracking method must be required. In the future, we will extend our method to design an ultrafast and robust feature tracking method for 3D reconstruction based on simultaneous localization and mapping.

Acknowledgement

This work is supported by grants from the National Key Research and Development Plan under Grant No. 2016YFC0800106, the National Natural Science Foundation of China (Nos. 61802103, 61877016, 61602146 and 61673157), the China Postdoctoral Science Foundation Grant No. 2018M632522, and the Fundamental Research Funds for the Central Universities (Nos. JZ2018HGBH0280 and PA2018GDQT0014).

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.optlastec.2018.08.045.

References

[1] N. Michael, M. Drakou, A. Lanitis, Model-based generation of personalized full-body 3D avatars from uncalibrated multi-view photographs, Multimedia Tools Appl. (2016) 1–27.
[2] G. Klein, D. Murray, Parallel tracking and mapping for small AR workspaces, Mixed and Augmented Reality, 2007. ISMAR 2007, in: 6th IEEE and ACM International Symposium on, IEEE, 2007, pp. 225–234.
[3] T. Kelly, P. Wonka, P. Wonka, N.J. Mitra, BigSUR: large-scale structured urban reconstruction, Acm Trans. Graph. 36 (6) (2017) 204.
[4] Z. Lu, P. Guerrero, N.J. Mitra, A. Steed, Open3D: crowd-sourced distributed curation of city models, in: Proceedings of the 21st International Conference on Web3D Technology, ACM, 2016, pp. 87–94.
[5] Y. Tian, C. Chen, M. Shah, Cross-view image matching for geo-localization in

Urban, Environments (2017).
[6] X. Xu, L. He, H. Lu, L. Gao, Y. Ji, Deep adversarial metric learning for cross-modal retrieval, World Wide Web (2018).
[7] L. He, X. Xu, H. Lu, Y. Yang, F. Shen, H.T. Shen, Unsupervised cross-modal retrieval through adversarial learning, in: 2017 IEEE International Conference on Multimedia and Expo (ICME), 2017, pp. 1153–1158.
[8] Y. Li, N. Snavely, D. Huttenlocher, P. Fua, Worldwide pose estimation using 3d point clouds, European Conference on Computer Vision (2012) 15–29.
[9] K. Sakurada, T. Okatani, K. Deguchi, Detecting changes in 3D structure of a scene from multi-view images captured by a vehicle-mounted camera, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 137–144.
[10] D. Wang, H. Lu, M.H. Yang, Robust visual tracking via least soft-threshold squares, IEEE Trans. Circuits Syst. Video Technol. 26 (9) (2016) 1709–1721.
[11] D. Wang, H. Lu, Z. Xiao, M.H. Yang, Inverse sparse tracker with a locally weighted distance metric, IEEE Trans. Image Process. 24 (9) (2015) 2646–2657.
[12] M. Colbert, J.-Y. Bouguet, J. Beis, S. Childs, D. Filip, L. Vincent, J. Lim, S. Satkin, Building indoor multi-panorama experiences at scale, ACM Siggraph 2012 Talks, ACM, 2012, p. 24.
[13] S. Song, M. Chandraker, Robust scale estimation in real-time monocular SFM for autonomous driving, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1566–1573.
[14] Q. Cui, V. Fragoso, C. Sweeney, P. Sen, GraphMatch: Efficient Large-Scale Graph Construction for Structure from Motion, arXiv preprint arXiv:1710.01602 (2017).
[15] S. Ramalingam, P. Sturm, A unifying model for camera calibration, IEEE Trans. Pattern Anal. Mach. Intell. 39 (7) (2017) 1309–1319.
[16] A. Chatterjee, V.M. Govindu, Robust relative rotation averaging, IEEE Trans. Pattern Anal. Mach. Intell. 99 (2017) 1–15.
[17] M.W. Cao, W. Jia, Y. Zhao, S.J. Li, X.P. Liu, Fast and robust absolute camera pose estimation with known focal length, Neural Comput. Appl. (2017).
[18] H. Lei, G. Jiang, L. Quan, Fast descriptors and correspondence propagation for robust global point cloud registration, IEEE Trans. Image Process. 26 (8) (2017) 3614–3623.
[19] L. Kang, L. Wu, Y.-H. Yang, Robust multi-view L 2 triangulation via optimal inlier selection and 3D structure refinement, Pattern Recogn. 47 (9) (2014) 2974–2992.
[20] M. Cao, S. Li, W. Jia, S. Li, X. Liu, Robust bundle adjustment for large-scale structure from motion, Multimedia Tools Appl. 76 (21) (2017) 21843–21867.
[21] H. Cui, X. Gao, S. Shen, Z. Hu, HSfM: Hybrid Structure-from-Motion, IEEE Conf. Comput. Vision Pattern Recogn. (2017) 2393–2402.
[22] J.L. Schönberger, J.-M. Frahm, Structure-from-motion revisited, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), CVPR, 2016, pp. 4104–4113.
[23] O. Ozyesil, V. Voroninski, R. Basri, A. Singer, A Survey of Structure from Motion (2017).
[24] C. Wu, S. Agarwal, B. Curless, S.M. Seitz, Multicore bundle adjustment, Computer Vision and Pattern Recognition (CVPR), in: 2011 IEEE Conference on, IEEE, 2011, pp. 3057–3064.
[25] D.J. Crandall, A. Owens, N. Snavely, D.P. Huttenlocher, SfM with MRFs: Discrete-Continuous Optimization for Large-Scale Structure from Motion, IEEE Trans. Pattern Anal. Mach. Intell. 35 (12) (2013) 2841–2853.
[26] B. Bhowmick, S. Patra, A. Chatterjee, V.M. Govindu, S. Banerjee, Divide and Conquer: Efficient Large-Scale Structure from Motion Using Graph Partitioning, Asian Conference on Computer Vision (2014) 273–287.
[27] C. Sweeney, V. Fragoso, T. Höllerer, M. Turk, Large Scale SfM with the Distributed Camera Model, 2016 Fourth International Conference on 3D Vision (3DV), 2016, pp. 230–238.
[28] M.R.U. Saputra, A. Markham, N. Trigoni, Visual SLAM and structure from motion in dynamic environments: a survey, ACM Comput. Surv. 51 (2) (2018) 1–36.
[29] G. Zhang, H. Liu, Z. Dong, J. Jia, T.T. Wong, H. Bao, Efficient Non-Consecutive Feature Tracking for Robust Structure-From-Motion, IEEE Trans. Image Process. 25 (12) (2016) 5957–5970.
[30] S.N. Sinha, J.-M. Frahm, M. Pollefeys, Y. Genc, GPU-based video feature tracking and matching, EDGE, Workshop on Edge Computing Using New Commodity Architectures, 2006, pp. 189–196.
[31] V. Garcia, E. Debreuve, F. Nielsen, M. Barlaud, K-nearest neighbor search: Fast GPU-based implementations and application to high-dimensional feature matching, in: IEEE International Conference on Image Processing, IEEE, 2010, pp. 3757–3760.
[32] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vision 60 (2) (2004) 91–110.
[33] J. Zhao, J. Ma, J. Tian, J. Ma, D. Zhang, A robust method for vector field learning with application to mismatch removing, 32(14) (2011) 2977–2984.
[34] S. Birchfield, Derivation of kanade-lucas-tomasi tracking equation, School of Computer Science, Carnegie Mellon University, Pittsburgh, 1997.
[35] B.D. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, Int. Joint Conf. Artific. Intell. (1981) 674–679.
[36] C. Tomasi, T. Kanade, Detection and tracking of point features, School of Computer Science, Carnegie Mellon Univ, Pittsburgh, 1991.
[37] G. Zhang, Z. Dong, J. Jia, T.-T. Wong, H. Bao, Efficient non-consecutive feature tracking for structure-from-motion, European Conference on Computer Vision, Springer, 2010, pp. 422–435.
[38] J. Jiang, A. Yilmaz, Good features to track: A view geometric approach, Computer Vision Workshops (ICCV Workshops), in: 2011 IEEE International Conference on, IEEE, 2011, pp. 72–79.
[39] T. Lee, T. Hollerer, Hybrid feature tracking and user interaction for markerless augmented reality, in: IEEE Conference on Virtual Reality, IEEE, 2008, pp. 145–152.
[40] B. Poling, G. Lerman, A. Szlam, Better feature tracking through subspace constraints, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 3454–3461.
[41] G. Zhang, P.A. Vela, Good Features to Track for Visual SLAM, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1373–1382.
[42] M. Garrigues, A. Manzanera, Real time semi-dense point tracking, International Conference Image Analysis and Recognition, Springer, 2012, pp. 245–252.
[43] S.N. Sinha, J.-M. Frahm, M. Pollefeys, Y. Genc, Feature tracking and matching in video using programmable graphics hardware, Mach. Vis. Appl. 22 (1) (2011) 207–217.
[44] J. Shi, C. Tomasi, Good features to track, IEEE onComputer Vision and Pattern Recognition, IEEE, 1994, pp. 593–600.
[45] K. Jia, T.-H. Chan, Z. Zeng, S. Gao, G. Wang, T. Zhang, Y. Ma, ROML: A robust feature correspondence approach for matching objects in a set of images, Int. J. Comput. Vision 117 (2) (2015) 1–25.
[46] M. Hwangbo, J.-S. Kim, T. Kanade, Inertial-aided KLT feature tracking for a moving camera, IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2009, pp. 1909–1916.
[47] K. Wilson, N. Snavely, Network principles for sfm: Disambiguating repeated structures with local context, Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 513–520.
[48] D. Ceylan, N.J. Mitra, Y. Zheng, M. Pauly, Coupled structure-from-motion and 3D symmetry detection for urban facades, ACM Trans. Graph. (TOG) 33 (1) (2014) 2.
[49] J. Engel, T. Schöps, D. Cremers, LSD-SLAM: Large-scale direct monocular SLAM, Computer Vision–ECCV, Springer 2014 (2014) 834–849.
[50] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, W. Burgard, An evaluation of the RGB-D SLAM system, Robotics and Automation (ICRA), in: 2012 IEEE International Conference on, IEEE, 2012, pp. 1691–1696.
[51] C. Peng, S. Sahani, J. Rushing, A GPU-accelerated approach for feature tracking in time-varying imagery datasets, IEEE Trans. Visual Comput. Graph. 23 (10) (2017) 2262–2274.
[52] M. Garrigues, A. Manzanera, Real time semi-dense point tracking, Image Anal. Recogn. (2012) 245–252.
[53] A. Buchanan, A. Fitzgibbon, Interactive feature tracking using kd trees and dynamic programming, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, 2006, pp. 626–633.
[54] G. Zhang, H. Liu, Z. Dong, J. Jia, T.-T. Wong, H. Bao, ENFT: Efficient Non-Consecutive Feature Tracking for Robust Structure-from-Motion, arXiv preprint arXiv:1510.08012 (2015).
[55] C. Wu, B. Clipp, X. Li, J.-M. Frahm, M. Pollefeys, 3d model matching with viewpoint-invariant patches (vip), in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2008, pp. 1–8.
[56] R. Raguram, O. Chum, M. Pollefeys, J. Matas, J. Frahm, Usac: A universal framework for random sample consensus, Pattern Anal. Mach. Intell., IEEE Trans. 35 (8) (2013) 2022–2038.
[57] J. Civera, O.G. Grasa, A.J. Davison, J.M.M. Montiel, 1-Point RANSAC for extended Kalman filtering: Application to real-time structure from motion and visual odometry, J. Field Rob. 27 (27) (2010) 609–631.
[58] C. Zach, M. Klopschitz, M. Pollefeys, Disambiguating visual relations using loop constraints (2010) 1426–1433.
[59] L. Svärm, Z. Simayijiang, O. Enqvist, C. Olsson, Point track creation in unordered image collections using Gomory-Hu trees, Pattern Recognition (ICPR), in: 2012 21st International Conference on, IEEE, 2012, pp. 2116–2119.
[60] R.E. Gomory, T.C. Hu, Multi-terminal network flows, J. Soc. Ind. Appl. Math. 9 (4) (1961) 551–570.
[61] M. Garrigues, A. Manzanera, T.M. Bernard, Video extruder: a semi-dense point tracker for extracting beams of trajectories in real time, J. Real-Time Image Proc. 11 (4) (2016) 785–798.
[62] L. Zhao, X. Li, J. Xiao, F. Wu, Y. Zhuang, Metric learning driven multi-task structured output optimization for robust keypoint tracking, Proceedings of the 29th {AAAI} Conference on Artificial Intelligence, 2015, pp. 3864–3870.
[63] N. Snavely, S.M. Seitz, R. Szeliski, Photo tourism: exploring photo collections in 3D ACM transactions on graphics (TOG), ACM (2006) 835–846.
[64] S. Agarwal, N. Snavely, I. Simon, S.M. Seitz, R. Szeliski, Building rome in a day, in: IEEE 12th International Conference on Computer Vision, IEEE, 2009, pp. 72–79.
[65] C. Zach, ETH-V3D Structure-and-Motion software.© 2010-2011, ETH Zurich (2010).
[66] Y. Gao, J. Luo, H. Qiu, B. Wu, Survey of structure from motion, Proceedings of 2014 International Conference on Cloud Computing and Internet of Things, 2014, pp. 72–76.
[67] Z. Dong, G. Zhang, J. Jia, H. Bao, Keyframe-based real-time camera tracking, in: IEEE 12th International Conference on Computer Vision, IEEE, 2009, pp. 1538–1545.
[68] C. Wu, SiftGPU: A GPU implementation of scale invariant feature transform, URL http://cs.unc.edu/~ccwu/siftgpu (2011).
[69] C. Wu, Towards linear-time incremental structure from motion, International Conference on 3D Vision-3DV, IEEE, 2013, pp. 127–134.
[70] Y. Furukawa, J. Ponce, Accurate, Dense, and Robust Multi-View Stereopsis, Cvpr, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2007, pp. 1362–1376.
[71] K. Ni, F. Dellaert, HyperSfM, 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), in: 2012 Second International Conference on, IEEE, 2012, pp. 144–151.
[72] P. Moulon, P. Monasse, R. Marlet, Global fusion of relative motions for robust,

accurate and scalable structure from motion, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 3248-3255.
[73] P.F. Alcantarilla, A. Bartoli, A.J. Davison, KAZE features, Computer Vision–ECCV, Springer 2012 (2012) 214–227.
[74] C. Sweeney, T. Sattler, T. Hollerer, M. Turk, M. Pollefeys, Optimizing the Viewing Graph for Structure-from-Motion, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 801–809.
[75] K. Wilson, N. Snavely, Robust global translations with 1dsfm, European Conference on Computer Vision, Springer, 2014, pp. 61–75.
[76] J. Xiao, A. Owens, A. Torralba, SUN3D: A database of big spaces reconstructed using sfm and object labels, Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 1625–1632.
[77] S.Y. Bao, S. Savarese, Semantic structure from motion, IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2011, pp. 2025–2032.
[78] T.Y. Wang, P. Kohli, N.J. Mitra, Dynamic SFM: Detecting scene changes from image pairsComputer Graphics Forum, Wiley Online Library, 2015, pp. 177–189.
[79] E. Zheng, C. Wu, Structure from Motion Using Structure-less Resection, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2075–2083.
[80] D. Crandall, A. Owens, N. Snavely, D. Huttenlocher, Discrete-continuous optimization for large-scale structure from motion, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2011, pp. 3001–3008.
[81] R. Arandjelović, A. Zisserman, Three things everyone should know to improve object retrieval, in: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE, 2012, pp. 2911–2918.
[82] X. Yang, K.-T. Cheng, LDB: An ultra-fast feature for scalable augmented reality on mobile devices, Mixed and Augmented Reality (ISMAR), in: 2012 IEEE International Symposium on, IEEE, 2012, pp. 49–57.
[83] O. Chum, J. Matas, Matching with PROSAC-progressive sample consensus, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, 2005, pp. 220–226.
[84] Z. Zhang, Q. Shi, J. McAuley, W. Wei, Y. Zhang, A. Van Den Hengel, Pairwise Matching through Max-Weight Bipartite Belief Propagation, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 16), To Appear, 2016.
[85] A. Knapitsch, J. Park, Q.-Y. Zhou, V. Koltun, Tanks and temples: benchmarking large-scale scene reconstruction, ACM Trans. Graph. 36 (4) (2017) 1–13.
[86] D. Mishkin, J. Matas, M. Perdoch, MODS: Fast and robust method for two-view matching, Comput. Vis. Image Underst. 141 (2015) 81–93.
[87] J. Heinly, E. Dunn, J.-M. Frahm, Comparative evaluation of binary features, Computer Vision–ECCV, Springer 2012 (2012) 759–773.