

## EXTRACTION OF IMAGE TOPOLOGICAL GRAPH FOR RECOVERING THE SCENE GEOMETRY FROM UAV COLLECTIONS

Zhihua Xu<sup>a</sup>, Lixin Wu<sup>b,\*</sup>, Shaojie Chen<sup>c</sup>, Ran Wang<sup>d</sup>, Fashuai Li<sup>d</sup>, Qiuling Wang<sup>a</sup>

<sup>a</sup> Key Laboratory of Environmental Change & Natural Disaster of MOE, Beijing Normal University, Beijing 100875, China - hzxurs@mail.bnu.edu.cn, wangqiuling\_2008@126.com

<sup>b</sup> IoT Perception Mine Research Center, China University of Mining and Technology, Xuzhou 221008, China – awulixin@263.net

<sup>c</sup> School of Resource Engineering, Longyan University, Longyan 364000, China – csj646@tom.com

<sup>d</sup> College of Geoscience and Surveying Engineering, China University of Mining and Technology, Beijing 100083, China - WangR0225@163.com, lifashuai@126.com

### Commission

**KEY WORDS:** CUDA-SIFT, Topological Graph, UAV Collections, Image Matching, 3D Geometry

### ABSTRACT:

This study was performed aiming to construct the scene geometry with a large set of unmanned aerial vertical (UAV) collections. By improving the popular structure from motion (SfM) algorithm, we focus on the efficiency improvement on procedures of both feature detection and image matching. Distinctive features are firstly detected with a CUDA based GPU accelerate technology under the basic of SIFT algorithm (CUDA-SIFT). And then, the image topological graph is computed by finding the conjunction relationship between UAV collections with the help of flight control data acquired by the UAV platform. Image matching will be guided by the computed image topological graph to solve the traversal matching problem. Experimental results show that CUDA-SIFT performs much better than the original SIFT algorithm on both efficiency and feature amount. Also, the topological graph of computed image limits the searching range for feature similarity computation, resulting in dramatic speed up. A final bundler adjustment is implemented in the procedure of scene geometry reconstruction, and the structural geometry as well as the coverage completeness is far more comparable to the SfM method.

## 1. INTRODUCTION

In the recent years, more and more applications of unmanned aerial vehicle system (UAVs) in the geomatics field became common. Aerial images acquired by a low attitude UAV that integrates high resolution, multi-view sensing, and large overlaps are useful for landscape surveying, disaster enhanced identification, as well as scene geometry reconstruction, etc. [1]-[3]. Image-based 3D modelling techniques have enabled creation of dense digital terrain models from a set of multi-view UAV optical collections [4]. This exciting development suggests the possibility of recovering the large scene from UAV optical sequences and video frames [5]-[7]. However, experimental study shows that the traditional method like the popular structure from motion (SfM) algorithm [8] usually emerge memory explosion when dealing with UAV collections which are of high resolution. Also, it is a computational procedure for the added images should search for all the candidates when matching images. The number of computations required to match feature points is quadratic in the amount of images, resulting in huge redundancy [9].

Scientists have made great efforts to solve the problems mentioned above. SfM scaling techniques like sub-sampling and hierarchical decomposition are suit for ordered video sequences that recover the scene geometry with sparse point cloud, which however may result in a low model accuracy [10]-[11]. Further attempt like GPS-based matching [1] roughly

position the images in space, thus limiting the set of possible matches. Identify the image conjunction with GPS only usually result in rough position because of both the positional accuracy and flight attitudes. Moreover, the GPU-based accelerate technology which is initially proposed to solve the big data problem are nowadays implemented to improve the efficiency of 3D modelling, and receives great achievements [12].

In our study, we improve the performance of scene geometry reconstruction with a CUDA-SIFT algorithm for feature detection, which is followed by an image topological graph guided strategy for image matching. The remainder of this paper is organized as follows. Section 2, the methodology is introduced specifically. The performance of our proposed method is presented in Section 3. Conclusion remarks and future work are addressed in Section 4.

## 2. METHODOLOGY

### 2.1 CUDA-SIFT

The Scale Invariant Feature Transform (SIFT) [13] is accepted as one of the best algorithms for distinctive feature detection. Distinctive features are identified by assigning orientations based on local image gradient directions. This procedure mainly consists of the following steps [13]:

1. *Scale-space extrema detection*: this is the stage where the keypoints are detected. For this, the image is convolved with

\* Corresponding author. Email: awulixin@263.net.

Gaussian filters at different scales, and then the differences of successive Gaussian-blurred images are taken. Keypoints are then taken as maxima/minima of the Difference of Gaussians (DoG) that occur at multiple scales. Specifically, a DoG image  $D(x, y, \sigma)$  is given by

$$D(x, y, \sigma) = L(x, y, k_1\sigma) - L(x, y, k_2\sigma) \quad (1)$$

Where  $L(x, y, k\sigma)$  is the convolution of the original image  $I(x, y)$  with the Gaussian blur  $G(x, y, k\sigma)$  at scale  $k\sigma$ ,

$$L(x, y, k\sigma) = G(x, y, k\sigma) * I(x, y) \quad (2)$$

2. *Keypoint localization*: after the procedure of scale-space extrema detection, too many keypoints candidates, some of which are unstable. The following step allows points to be rejected that have low contrast which are sensitive to noise or are poorly localized along an edge.

3. *Orientation assignment*: in this step, each keypoint is assigned one or more orientations based on local image gradient directions. First, the Gaussian-smoothed image  $L(x, y, \sigma)$  at the keypoint's scale  $\sigma$  is taken so that all computations are performed in a scale-invariant manner. For an image sample  $L(x, y)$  at scale  $\sigma$ , the gradient magnitude,  $m(x, y)$ , and orientation,  $\theta(x, y)$ , are precomputed using pixel differences:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad (3)$$

$$\theta(x, y) = \tan^{-1} \left( \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)} \right) \quad (4)$$

4. *Keypoint descriptor*: this step is performed to compute a descriptor vector for each keypoint which is highly distinctive and partially invariant to the remaining variations such as illumination, 3D viewpoint, etc. The descriptor is formed from a vector containing the values of all the orientation histogram, which are computed from magnitude and orientation values of samples in a  $16 \times 16$  region around the keypoint such that each histogram contains samples from a  $4 \times 4$  subregion of the original neighborhood region. The magnitudes are further weighted by a Gaussian function with  $\sigma$  equal to one half the width of the descriptor window. The descriptor then becomes a vector of all the values of these histograms containing 128 elements.

Our experimental texts show that the procedure of scale-space extrema detection is much more time consuming, taking about 30% to 50% of the whole workflow, keypoint localization takes about 10% to 20%, keypoint descriptor takes about 20% to 30%, while the orientation assignment takes the least.

For better performance of the SIFT algorithm to deal with the UAV collections with high resolution, we focus on the following improvements. Fig.1 illustrates the workflow of our proposed CUDA-SIFT algorithm. Firstly, the UAV images are divided into blocks of equal size, leaving common features between blocks nearby, which will help to escape the problem of memory explosion. This procedure is conducted in the CPU memory, requiring little cost. And then, image blocks together with Gaussian parameters are delivered to GPU as texture memory and constant memory, separately. All the images blocks are processed with a same Gaussian parameter, as stored in the constant memory. Next, the blocks stored in the texture memory are further processed by individually detecting the distinctive features on the basic of SIFT algorithm, generating keypoint descriptors as outputs in each image block. Finally, keypoints in each block are transported back to the CPU memory and to be merged into a common coordinate with the pre-defined pixel indexes to generate the final keypoint file.

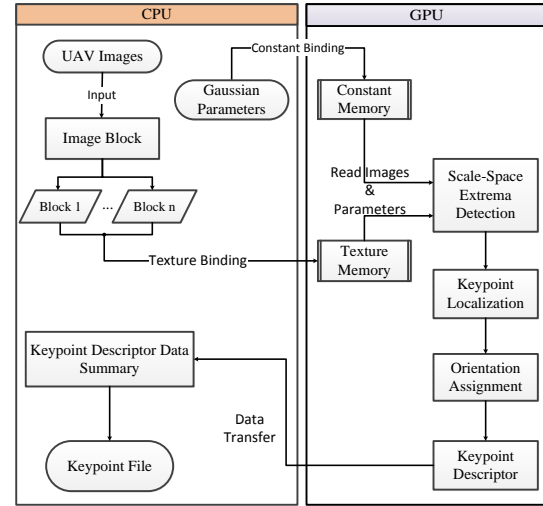


Fig. 1 Procedure of feature detection with CUDA-SIFT algorithm

## 2.2 Image Topological Graph

**2.2.1 Flight control data acquired by the UAV system:** with the development of GNSS/INS systems, it is necessary to navigate the UAV to the predefined acquisition points, recording the position and orientation of the platform at the time of camera exposure. Although the precision of navigation is not as accuracy as high quality GNSS/INS devices, it is good enough to help to identify the conjunction relationship between images nearby.

**2.2.2 Image Topology Analysis:** Image topology is used to identify the conjunction relationship between UAV images and indexed for image matching. It is conducted through relative orientation with the support of flight-control data acquired by UAV platform. This procedure contains of the following steps.

**Step1.** Transform the flight control data. Except for the position information, flight control system also records the attitude angle sensor for innovative marketing body coordinate system in the navigation of roll, pitch and yaw  $(\Phi, \Theta, \Psi)$ , which should be transformed to the photographic external angles  $(\psi, \omega, \kappa)$  for user coordinate in the pixel space coordinate.

**Step 2.** Compute the coordinates of the four vertexes for each of the UAV image through relative orientation referring to formula (5), (6).

$$X_{i,j} = h_{i,0} \cdot \frac{(a_{i,1} \cdot Vx_{i,j} + b_{i,1} \cdot Vy_{i,j} - c_{i,1} \cdot f_i)}{(a_{i,3} \cdot Vx_{i,j} + b_{i,3} \cdot Vy_{i,j} - c_{i,3} \cdot f_i)} + X_{i,0} \quad (5)$$

$$Y_{i,j} = h_{i,0} \cdot \frac{(a_{i,2} \cdot Vx_{i,j} + b_{i,2} \cdot Vy_{i,j} - c_{i,2} \cdot f_i)}{(a_{i,3} \cdot Vx_{i,j} + b_{i,3} \cdot Vy_{i,j} - c_{i,3} \cdot f_i)} + Y_{i,0} \quad (6)$$

Where,  $(X_{i,j}, Y_{i,j})$  represents the coordinate of vertex  $j$  in image  $i$ ,  $h_{i,0}$  represents the relative height above the ground and  $(X_{i,0}, Y_{i,0})$  shows the position of camera  $i$ ,  $(Vx, Vy)$  indicates the vertex coordinates in the pixel space, and  $(a, b, c)$  are the external parameters.

**Step 3.** Identify the topological relationship between one image and each vertex of other images. The topology between image  $P$  and vertex  $V$  is identified by computing the area of all the triangles composed with the vertex  $V$  and any other two vertex of  $P$ , as shown in Fig. 2, and the topology can be identified with formula (7).

$$T(P, V) = \begin{cases} 1, S_P = S_{\Delta V_{P12}V} + S_{\Delta V_{P23}V} + S_{\Delta V_{P34}V} + S_{\Delta V_{P41}V} \\ 0, S_P \neq S_{\Delta V_{P12}V} + S_{\Delta V_{P23}V} + S_{\Delta V_{P34}V} + S_{\Delta V_{P41}V} \end{cases} \quad (7)$$

Where,  $T(P, V) = 1$  means the vertex  $V$  is contained in image  $P$ , while  $T(P, V) = 0$  indicates the vertex  $V$  lies outside of image  $P$ .

**Step 4.** Identify the topology between any pair of images. If none of the vertexes from any two images lies in each other, there are no common features between this image pair. The image topology is identified as formula (8).

$$T(P_i, P_l) = \begin{cases} 1, \sum_{j=1}^4 T(P_i, V_{ij}) + \sum_{j=1}^4 T(P_l, V_{ij}) \neq 0 \\ 0, \sum_{j=1}^4 T(P_i, V_{ij}) + \sum_{j=1}^4 T(P_l, V_{ij}) = 0 \end{cases} \quad (8)$$

Where,  $T(P_i, P_l)$  is the topological relationship between image  $P_i$  and  $P_l$ .

**Step 5.** Compute the topological graph of the UAV collections. Specifically, we consider the topological collections an image graph, with a node for every image, and two directed edges between any pair of images with common features. The graph is illustrated by point sets as  $V(G)$ , edge sets as  $E(G)$  as follows:

$$V(G) = \{P_1, P_2, P_3, \dots, P_n\} \quad (9)$$

$$E(G) = \{\{P_i, P_j\} | (T(P_i, P_j) = 1, 1 \leq i \leq n, 1 \leq j \leq n)\} \quad (10)$$

Where,  $n$  means the number of the collections.

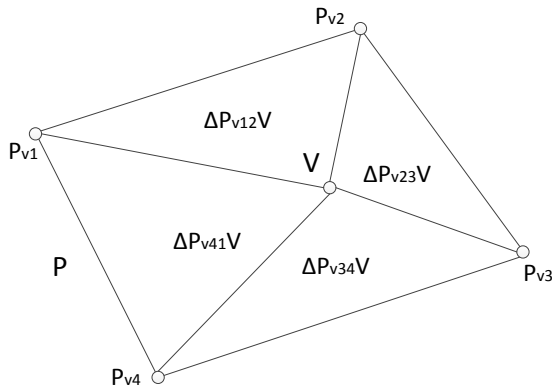


Fig. 2 Sketch map for image-vertex topology identification

### 2.3 Image Matching

For image matching, CUDA-SIFT features are first extracted from a set of reference images and stored in a database. A new image is matched by individually comparing each feature from the new image to this predetermined topological conjunction images and finding candidate matching features based on Euclidean distance of their feature vectors. The smaller the Euclidean distance the more likely the two points correspond [13]. That will results in quite many wrong correspondences, from which we can further identify the feasible ones.

$$f_{mn} = \arg \min \|f_d(n) - f_d(c)\|_2 \quad (11)$$

Where  $f_{mn}$  is the minimum distance of potential common feature between the new image and the candidate image, whose Euclidean distance is  $f_d(n)$  and  $f_d(c)$ , separately.

For each point in the first image of an image pair we search for the best and second best candidates in the other image. In case of the ratio  $r = d_{second} / d_{best}$  of the certain measures for the second best to the best candidate is lower than a certain threshold, here  $r < 0.6$ , we accept it as feasible correspondence.

**2.3.1 Complexity analysis:** to simply the case, we assume that all the images are with the maximum topology, as  $k$ . At the worst situation, the current image will match all the candidates when its index number is smaller than  $k$ . And other images, whose indexes are larger than  $k$  will match their corresponding  $k$  images only. Then the matching complexity will be formulized on image level as:

$$\frac{k \cdot (k-1)}{2} + \sum_k^n k = kn - \frac{k^2 + k}{2} \quad (12)$$

Where,  $k$  represents the maximum image topology and  $n$  means the number of UAV collections.

### 2.4 Bundler Adjustment

After the procedure of CUDA-SIFT matching, the Least Square Matching (LSM) is implemented to refine the correspondences for sub-pixel localization and to avoid inaccurate matches. Bundler adjustment is implemented with open source code, like Bundler [9], which is used here to automatically orient the images to finally recover the camera parameters and the scene geometry.

## 3. RESULTS AND DISCUSSION

Fig. 3 illustrates a sample of the comparison of feature matching with keypoint descriptors between SIFT and the proposed CUDA-SIFT algorithms. Corresponding features detected with both CUDA-SIFT and SIFT algorithms are fine matched, getting a stable result. As the difference-of-Gaussian function has a strong response along edges, more CUDA-SIFT features lie in the block edges are additionally detected compared with SIFT features. About 925 corresponding features are fine matched between the image pair with CUDA-SIFT, giving better performances in details than that of SIFT, with which there are 844 corresponding features fine matched.

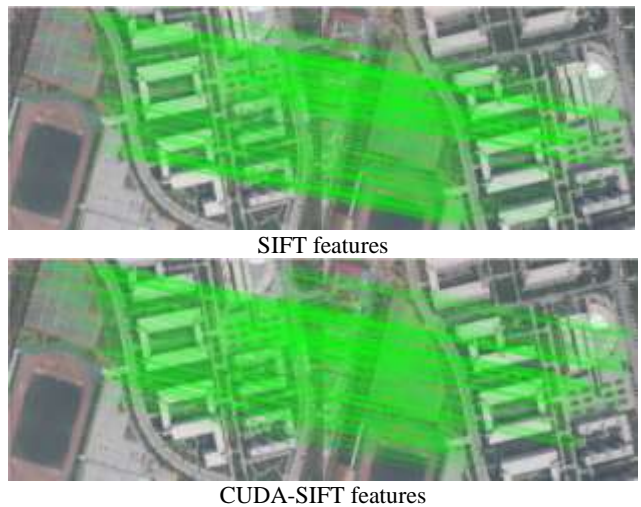


Fig. 3 Comparison of the matching performances between SIFT and the proposed CUDA-SIFT algorithms

To evaluate the efficiency of the CUDA-SIFT for feature detection, we compare its cost with that of SIFT algorithm by testing six data sets with different resolution and coverage complexity. For each data set we choose 10 images for feature detection. Results show that the performance of CUDA-SIFT keeps steady, which is little affected by the image resolution, whereas the cost of SIFT algorithm increases with image resolution, as indicted in Fig. 4. Fig. 5 displays the ratio of time consuming for SIFT to CUDA-SIFT in the process of feature detection from data sets motioned above. We can see that, the ratio increases from about 10 to 20 with image resolution from 2560×1920 to 5616×3744 pixels.

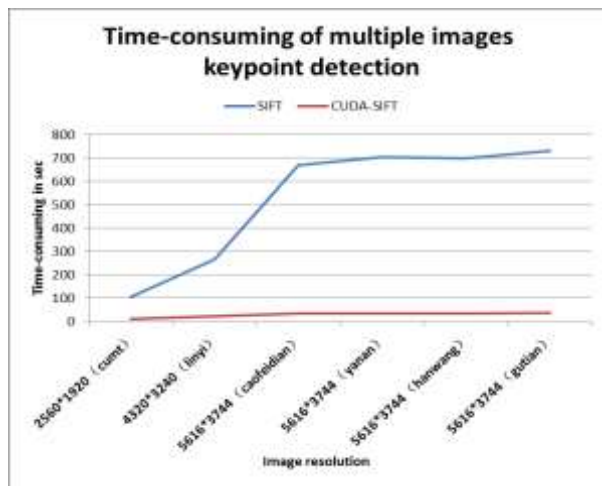


Fig. 4 Time-consuming of multiple images feature detection

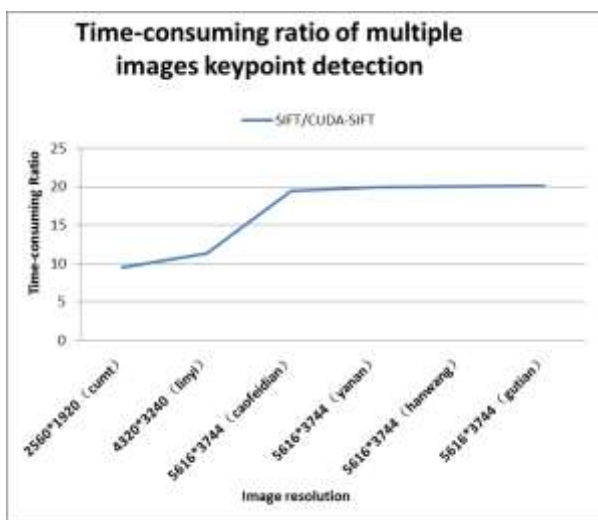


Fig. 5 Time-consuming ratio of multiple images feature detection

To evaluate the performance of the image topological graph based matching strategy for large scene geometry recover from UAV optical collections, a flight experiment with a fixed-wing UAV was conducted at Caofeidian in Hebei Province of China. The flight was about 300 meters high above the ground, and be affected by level 5 wind speed, resulting in four irregular lines, covering 72 images taken with Canon EOS 5D Mark II. The image resolution is about 5cm×5cm, with the size of 5616×3744 pixels.

The image graphs generated with both the traversal matching strategy and the conjunction relationship are illustrated in Fig. 6 and Fig. 7, separately. Here, the images are simplified as points,

and any pair of images with common features is connected with a direct edge. The image topological graph looks sparser that helps to reduce the matching images to 1357 from the traversal strategy of 2556, improving approximate twice of the matching cost.

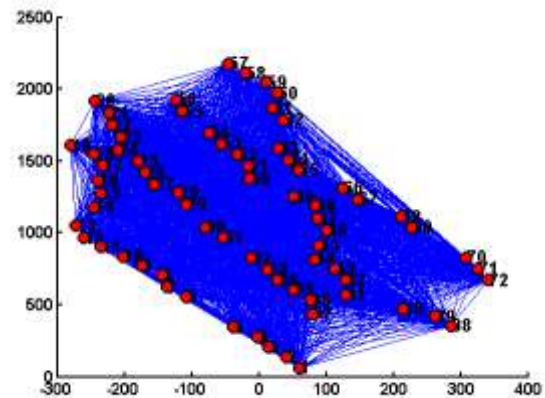


Fig. 6 Image graph with a traversal matching strategy

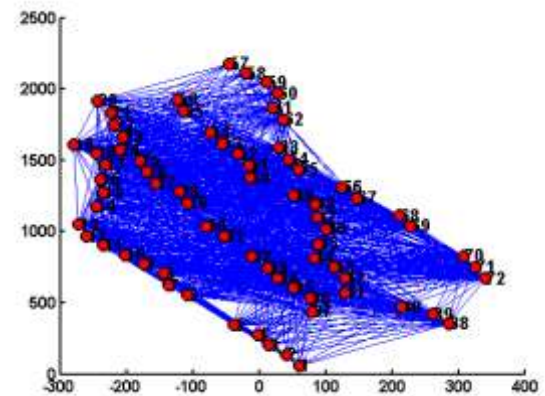


Fig. 7 Image topological graph computed with the conjunction relationship

Evidently, for large, redundant collections, a much smaller set of images are sufficient to represent most of the information about the scene [14]. Consequently, with the image topology, further attempts could be taken to compute a reliable and skeletal graph that helps to recover the main information with minimum images. Our previous study has done some attempt to compute the image topology skeleton, leaving huge space to improve, further information is illustrated in [15].

The sparse 3D model of the study area is recovered with our proposed method, as shown in Fig. 8. The geometry of the main objects that contain stock dumps, vehicles as well as landscape with complex texture are well reconstructed, while objects with lower texture contrast and monotonous coverage are failed to be recovered, which is resulted from the limited features that could be used for matching. This study area is also reconstructed through a traversal matching strategy with open source, like Bunder that requires only images as input. The computed scene geometry is shown in Fig. 9, displaying little apparent difference from that of our proposed method on neither the structural geometry nor the coverage completeness, which on the other hand verifies the effectiveness of image topological graph.



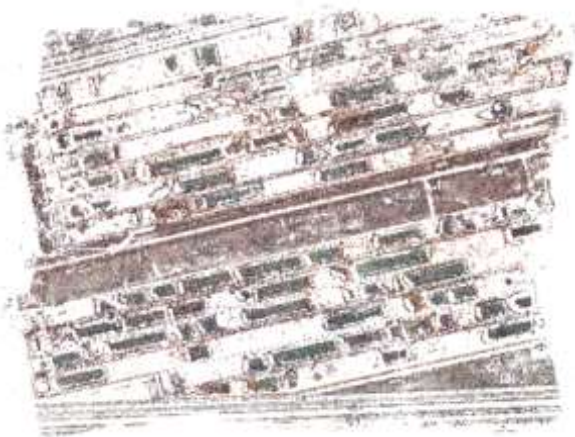


Fig. 8. Sparse scene geometry of the study area recovered with image topology matching strategy

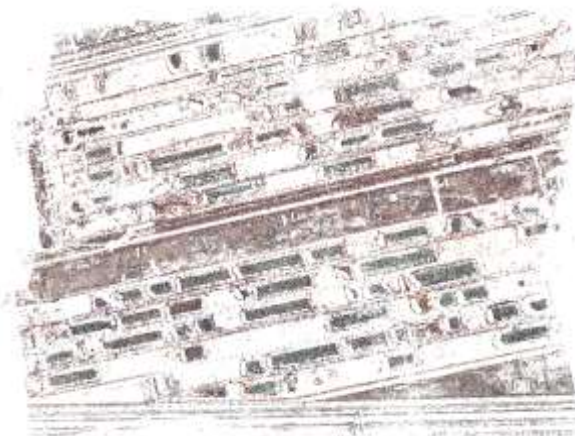


Fig. 9. Sparse scene geometry of the study area recovered with traversal matching strategy

#### 4. CONCLUSION

In this paper, we have proposed a CUDA-SIFT algorithm and conducted an image topological graph for improving the efficiency of scene geometry reconstruction. Various data sets with different resolution are used for CUDA-SIFT verification. The results verify its effectiveness for distinctive feature detection from UAV high resolution optical images, and also for the efficiency improvement and memory feasibility strengthen. A flight experiment is conducted with a fix-wing UAV, which is controlled automatically with a simple GPS/IMU system. And the image topological graph is computed by analysing the conjunction relationship between images nearby, as guidance for image matching. Experimental results show that, the image topological graph definitely limits the search range of images for similarity calculation between feature vectors, improving the matching efficiency dramatically, without losing completeness. For more efficiency, we may reduce the number of matching images together with the optimization of flight plan as next steps.

#### 5. REFERENCES

[1] Douterloigne, K., Gautama, S., Philips, W., 2010. On the accuracy of 3D landscapes from UAV image data. *IGARSS*, pp. 589-592.

- [2] Xu, Z. H., Liu, C. B., Wang, P., et al., 2013. Airborne union observation and disaster enhanced identification of  $M_s$  7.0 Lushan earthquake. *Science and Technology Review*, 31(12), pp: 37-41
- [3] Irschara, A., Kaufmann, V., Klopschitz, M., et al, 2010. Towards fully automatic photogrammetric reconstruction using digital images taken from UAVs. In *Proceedings of the ISPRS TC VII Symposium—100 Years ISPRS*, Vienna, Austria.
- [4] Rosnell T., Honkavaara E., 2012. Point cloud generation from aerial image data acquired by a Quadcopter type micro unmanned aerial vehicle and a digital still camera, *Sensors*, 12(1), pp. 453-480.
- [5] Turner D., Lucieer A., 2013. Using a micro unmanned aerial vehicle (UAV) for ultra high resolution mapping and monitoring of landslide dynamics, *IGARSS*, Melbourne, Australia.
- [6] Niethammer U., James M.R., Rothmund S. et al, 2012. UAV-based remote sensing of the Super-Sauze landslide: evaluation and results, *Engineering Geology*, 128(1), pp. 2-11.
- [7] Turner D., Lucieer A., Watson C., 2012. An automated technique for generating georectified mosaics from ultra-high resolution unmanned aerial vehicle (UAV) imagery, based on structure from motion (SfM) point clouds. *Remote Sensing*, 4(5), pp. 1392-1410.
- [8] Snavely, N., 2008. Scene reconstruction and visualization from Internet photo collections, University of Washington.
- [9] Snavely, N., 2010. Bundler: structure from motion (SfM) for unordered image collections. from <http://phototour.cs.washington.edu/bundler>
- [10] Lourakis M., and Argyros A., 2004. The design and implementation of a generic sparse bundle adjustment software package based on the levenberg-marquardt algorithm," *Institute of Computer Science-FORTH*, Greece.
- [11] Niethammer U., James M.R., Rothmund S., et al, 2012. UAV-based remote sensing of the Super-Sauze landslide: evaluation and results, *Engineering Geology*, vol. 128, pp. 2-11.
- [12] Wu C., 2013. Towards linear-time incremental structure from motion [C], *3DV*, Seattle, WA, USA.
- [13] Lowe, D. G., 2004. Distinctive image features from scale-invariant keypoints, *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 91-110.
- [14] Snavely N., Seitz S. M., Szeluski R., 2008. Skeletal graphs for efficient structure from motion," In: *CVPR*, pp. 1-8.
- [15] Xu Z. H. Wu L. X., Wang Z., et al., 2013. Matching UAV images with image topology skeleton, *IGARSS*, Melbourne, Australia, pp. 546-549.

#### 6. ACKNOWLEDGEMENT

We acknowledge financial support from National Basic Research Program of China under Grant (NO.2011CB707102) and Fundamental Research Funds for Central Universities under Grant (NO.105565GK).