Skeletal camera network embedded structure-from-motion for 3D scene reconstruction from UAV images

CrossMark

Zhihua Xu [a,b,c], Lixin Wu [d,*], Markus Gerke [c], Ran Wang [e], Huachao Yang [d]

[a] College of Geoscience and Surveying Engineering, China University of Mining and Technology (Beijing), Beijing 100083, China
[b] Key Laboratory of Environment Change & Natural Disaster, Ministry of Education, Beijing Normal University, Beijing 100875, China
[c] Faculty of Geo-Information Science and Earth Observation (ITC), University of Twente, 7500 AE Enschede, The Netherlands
[d] School of Environment Science & Spatial Informatics, China University of Mining and Technology, Xuzhou 221116, China
[e] School of Geography, Beijing Normal University, Beijing 100875, China

ABSTRACT

Structure-from-Motion (SfM) techniques have been widely used for 3D scene reconstruction from multi-view images. However, due to the large computational costs of SfM methods there is a major challenge in processing highly overlapping images, e.g. images from unmanned aerial vehicles (UAV). This paper embeds a novel skeletal camera network (SCN) into SfM to enable efficient 3D scene reconstruction from a large set of UAV images. First, the flight control data are used within a weighted graph to construct a topologically connected camera network (TCN) to determine the spatial connections between UAV images. Second, the TCN is refined using a novel hierarchical degree bounded maximum spanning tree to generate a SCN, which contains a subset of edges from the TCN and ensures that each image is involved in at least a 3-view configuration. Third, the SCN is embedded into the SfM to produce a novel SCN-SfM method, which allows performing tie-point matching only for the actually connected image pairs. The proposed method was applied in three experiments with images from two fixed-wing UAVs and an octo-copter UAV, respectively. In addition, the SCN-SfM method was compared to three other methods for image connectivity determination. The comparison shows a significant reduction in the number of matched images if our method is used, which leads to less computational costs. At the same time the achieved scene completeness and geometric accuracy are comparable.

© 2016 Published by Elsevier B.V. on behalf of International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS).

1. Introduction

3D scene reconstruction plays an important role in heritage documentation (El-Hakim et al., 2004), urban planning (Verdie et al., 2015), virtual reality (Feng et al., 2014), and disaster management (Bulatov et al., 2014; Ferworn et al., 2011; Gerke and Kerle, 2011). Many studies have investigated the benefits of Structure-from-Motion (SfM) techniques for 3D scene reconstruction from multi-view images (Pollefeys et al., 1999; Scaramuzza et al., 2006; Sturm and Triggs, 1996). Nowadays, many commercial and open source software packages integrating a SfM method are available (Koutsoudis et al., 2014). Unmanned aerial vehicles (UAV), showing great advantages in operational cost and flexibility, have been increasingly used to date to capture multi-view images (Colomina and Molina, 2014; Lucieer et al., 2014; Zhang and Kovacs, 2012). In particular, UAV images which are sequentially acquired with high geometrical resolution and large overlap, are suitable for 3D scene reconstruction. Many studies have investigated the use of UAV images and SfM methods for 3D reconstruction in landslides (Niethammer et al., 2012), agriculture (Dandois and Ellis, 2013), topography (Woodget et al., 2015), and disaster scenarios (Vetrivel et al., 2015).

In the literature, many efforts have been undertaken to reduce the computational cost associated especially to image matching and bundle adjustment within SfM. In this work, the attention is mainly devoted to image matching, one of the most time-consuming phases. An effective approach for computational savings is to embed a topologically connected camera network (TCN) within SfM as a constraint (Rupnik et al., 2013; Xu et al., 2014a). A TCN is also referred to as image connectivity graph, which identifies the connections between the images (Snavely et al., 2010). The TCN-embedded SfM allows only the connected image pairs to be involved in matching. This method has been successfully used for processing UAV images in both nadir and oblique

* Corresponding author.
  E-mail addresses: hzxurs@mail.bnu.edu.cn (Z. Xu), awulixin@263.net (L. Wu), m.gerke@utwente.nl (M. Gerke), wangr0225@163.com (R. Wang), 932400012@qq.com (H. Yang).

views. For instance, Douterloigne et al. (2010) utilized Global Navigation Satellite System (GNSS) data to generate a TCN for matching nadir UAV images. Rupnik et al. (2015) used the on-board sensors associated with GNSS and an Inertial Measurement Unit (IMU) to generate a TCN for dealing with oblique images. Moreover, a rough point cloud-derived TCN was employed for matching unordered UAV images or video frames (Alsadik et al., 2015; PhotoScan, 2013).

Many methods for generating a TCN are decision-making based, either on image footprint conjunction (Douterloigne et al., 2010; Rupnik et al., 2013) or on tie-points distribution (Alsadik et al., 2015; Castillo et al., 2015; PhotoScan, 2013). The major shortcoming of the mentioned approaches is that they operate on a pure direct adjacency principle, e.g. a minimum overlap between images. This, however, does not consider the entire image topology and may lead to adding matching pairs which are not relevant for the network, but cause additional matching cost, or it may lead to a loss of relevant image pairs, respectively.

In the literature, a spanning tree has been commonly used for optimally removing edges which are considered non-essential from a weighted graph (Held and Karp, 1970; Helmi and Rahmani, 2014). In particular, the extraction of the skeletal camera network (SCN) follows a maximum spanning tree (MST) approach (Gavril, 1987), which find a subset of edges from a weighted TCN to achieve a maximal total weight (i.e. overlap) with a minimally required number of edges. For a wide range of applications, several variants of MST have been carried out with maximum connectivity and degree bounded considerations (Gouveia et al., 2014; Helmi and Rahmani, 2014; Katagiri et al., 2012).

This paper aims to construct a SCN to improve the computational efficiency of SfM for 3D scene reconstruction from a large set of highly overlapping UAV images. Specifically, we first construct a weighted TCN to identify the connections between UAV images, then extract a SCN by deleting the non-essential edges from the TCN, and finally embed the SCN into a SfM method (SCN-SfM) in order to restrict pairwise matching to images which are connected in the SCN. This paper is organized as follows: Section 2 starts by describing the framework of the proposed method, followed by illustrations of each component of the proposed method and baseline methods used for comparison; Section 3 describes the experimental sites and data used; the experimental results are present in Section 4, followed by discussion in Section 5 and conclusions of this study in Section 6.

2. Methodology

The framework of the proposed SCN-SfM method for 3D scene reconstruction from UAV images is sketched in Fig. 1. It contains three main components, namely (a) TCN construction, (b) SCN extraction, and (c) SCN embedded SfM. We first construct a TCN to represent the dataset by a weighted graph, considering the connections between UAV images, using the flight control data. Subsequently, we obtain an optimal SCN by iteratively deleting the non-essential edges in the TCN using a novel spanning tree, i.e. the hierarchical degree bounded maximum spanning tree (HDB-MST). Finally, the optimal SCN is embedded into the SfM method for 3D scene reconstruction, where only the connected image pairs represented by the SCN are considered for tie-point matching. The three components of the proposed method are described in detail in the following sections.

2.1. TCN construction

We use the flight control data to construct a weighted TCN representing the connections between the images in a given UAV

collection (Rupnik et al., 2013; Xu et al., 2014a). For n images, we define the corresponding TCN by a weighted directed graph $G = (V, E)$ with its node set $V = \{v_1, \ldots, v_n\}$ and the edge set $E = \{e_{i,j} : i, j = 1, \ldots n\}$. Each node represents one image, and the edge represents a connected image pair. We represent the graph by a triangular adjacency matrix, in which we write $c_{i,j} = 1$ to denote $e_{i,j} \in E$, and we use the value of $e_{i,j}$ to denote the edge weight between a pair of nodes (v_i, v_j). Here, the edge weight is characterized by the overlap between an image pair. For the TCN construction, the footprints of each image are first computed from flight control data and by projecting the image outline onto an elevation model (Rupnik et al., 2013; Xu et al., 2014a). Next, the connections between the image pairs are identified through an image topology analysis (Xu et al., 2014a; Xu et al., 2015), followed by the calculation of overlap between the connected images. Fig. 2 illustrates five main categories of an overlapping image pair (v_1, v_2). The overlap of a connected image pair can be estimated by the following procedure: given a connected image pair (v_1, v_2), we first find the convex hull of the overlapping area $e_{1,2}$ by means of Graham scan algorithm (Kong et al., 1990), then divide the convex hull into multiple triangles and finally estimate the overlap by Eqs. (1) and (2).

$$e_{1,2} = \sum_{i=2}^{m-1} S_{\Delta t_1 t_i t_{i+1}} \tag{1}$$

where $t_i = (x_i, y_i)$ is the ith of m nodes of the convex hull, and $S_{\Delta t_1 t_i t_{i+1}}$ is the area of one triangle composed by node set $\{t_1, t_i, t_{i+1}\}$.

$$S_{\Delta t_1 t_i t_{i+1}} = \frac{1}{2} |(x_i - x_1)(y_{i+1} - y_1) - (x_{i+1} - x_1)(y_i - y_1)| \tag{2}$$

The use of the K-nearest-neighbors algorithm (Cover and Hart, 1967) can help to find corresponding overlapping images since the number of overlapping candidates is quadratic in the number of images.

Fig. 3 shows the weighted TCN of an example UAV collection with 11 images. Note that the information on the diagonal and the last column are auxiliary information described in the next subsections.

2.2. SCN extraction

The SCN minimizes the number of edges in TCN. We consider a representation in a graph $G = (V, E)$. The minimization is conducted using the proposed maximum spanning tree, i.e. HDB-MST. The application of HDB-MST is achieved by using two main procedures associated with *hierarchical TCN representation* and *SCN extraction from the hierarchically structured TCN* in an iterative configuration. The execution of the proposed HDB-MST is summarized by the following pseudo-codes:

```
while do
    let G = (V, E) be a weighted graph that represent TCN
    hierarchically restructure TCN and divide G = (V, E) into a
    set of n subgraphs as discussed in Section 2.2.1
    extract SCN from the hierarchically structured TCN as
    discussed in Section 2.2.2
    if SCN equals to TCN then
        exit
    else
        update TCN by SCN
    end if
end while
return SCN
```
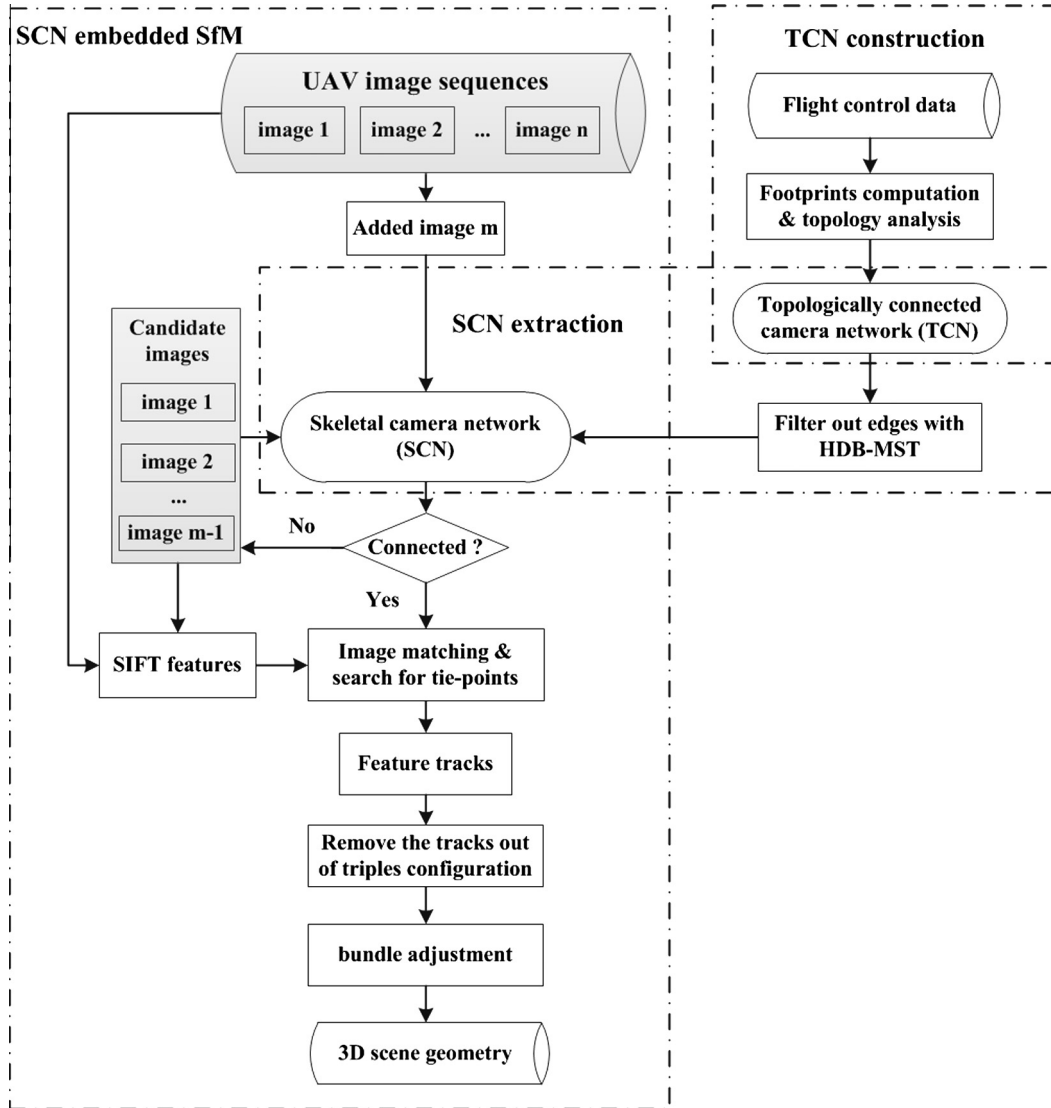
Fig. 1. Framework of the proposed SCN-SfM method for 3D scene reconstruction from UAV images.
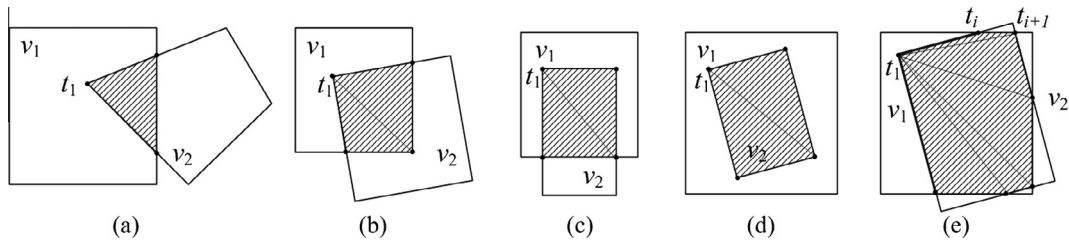

Fig. 2. Categories of overlapping image pair (v_1, v_2).

The details of the two components involved in the proposed HDB-MST algorithm are summarized in the following subsections. Let $\text{Adj}(v_i)$ denote the adjacent set of a node $v_i \in V$, $\text{Adj}(v_i) \triangleq \{v_j : c_{i,j} = 1 \cup c_{j,i} = 1\}$. Let $\#(S)$ denote the number of elements in the set S. Let $d(v_i)$ denote the degree of a node v_i, characterized by the number of edges incident with v_i as $d(v_i) = \#(\text{Adj}(v_i))$. Let $o(v_i)$ denote the totally accumulative edge weight between a node v_i and its adjacent set as $o(v_i) = \sum_{i,j}^{e}, v_j \in \text{Adj}(v_i)$. Let $w(v_i)$ denote the importance weight of a node v_i as:

$$w(v_i) = \frac{d(v_i) \cdot o(v_i)}{\sum\limits_{v_i \in V} d(v_i) \cdot o(v_i)} \tag{3}$$

Hence, we define a subgraph $G_i = (V_i, E_i)$, where the node set V_i consists of a seed $v_s \in V$ and its adjacent set $\text{Adj}(v_s)$, and the edge set $E_i = \{e_{t,r} : v_t, v_r \in V_i, c_{t,r} = 1\}$.

2.2.1. Hierarchical TCN representation

The hierarchical TCN representation assigns the node set $V = (v_1, \ldots, v_n)$ of the corresponding graph $G = (V, E)$ into multiple layers. It divides graph G into n subgraphs and specifies the order of each subgraph for processing. The achieved subgraph set is G^H and its node set is V^H. A temporary node set V^T is used to store the nodes in the layers from top to down; and the node set belonging to the lth layer is represented as V^l. The implemen-
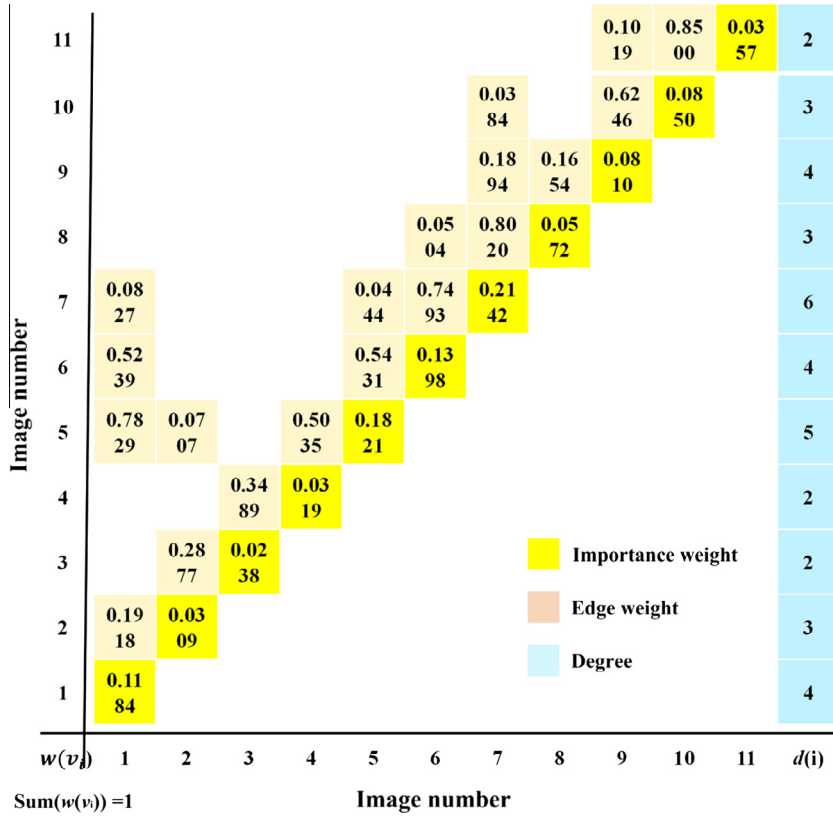
Fig. 3. The weighted TCN of an example UAV collection containing 11 images. The weighted TCN is represented by a modified $n \times (n+1)$ adjacency matrix ($n = 11$), where the data in the last column (in light blue) represents the degree $d(v_i)$ of node $v_i \in V$, and the value in the diagonal (in yellow) represents the importance weight $w(v_i)$ of node v_i. The remaining value (in light brown) represents the edge weight $e_{i,j}$ between a node pair (v_i, v_j). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

tation of the hierarchical TCN representation is summarized as follows:

Step 1: (Initialization) set $V^T \leftarrow \emptyset$, $l \leftarrow 1$, $V^l \leftarrow \emptyset$, $G^H \leftarrow \emptyset$ and $V^H \leftarrow \emptyset$

Step 2: select the node v_s with maximum importance weight in G as a seed and construct the first subgraph $G_1 = (V_1, E_1)$, where $V_1 = \{v_s, \text{Adj}(v_s)\}$; set $V^l \leftarrow \{v_s\}$, $V^{l+1} \leftarrow \{\text{Adj}(v_s)\}$, $V^T \leftarrow \{V^l, V^{l+1}\}$, $G^H \leftarrow \{G_1\}$, $V^H \leftarrow \{V_1\}$ and $l \leftarrow l + 1$

Step 3: (Seed extension) sort $w(V^l)$ in decreasing order
for $j = 1 : \#(V^l)$ do
  treat the jth node in the sorted node set V^l as a seed and construct the corresponding subgraph G_j in a similar manner; set $G^H \leftarrow \{G^H, G_j\}$, $V^H \leftarrow \{V^H, V_j\}$ and $j \leftarrow j + 1$
end for

Step 4: (Termination test) if $\#(G^H) = n$, go to Step 5; else $l \leftarrow l + 1$, $V^l \leftarrow \{\cup V^H \setminus V^T\}$, $V^T \leftarrow \{\cup V^H\}$ and go to Step 3.

Step 5: (Complete subgraph generation) return complete set of subgraphs G^H in well-organized order.

In Step 3, if there is more than one node with the same weight, we treat the respective nodes to be selected as seeds based on their orders in the node set V. Table 1 gives the components of the subgraphs after the hierarchical representation of TCN associated with the example weighted graph in Fig. 3.

2.2.2. SCN extraction from the hierarchically structured TCN

We formulate our problem by gradually filtering out the nonessential edges from each subgraph according to its order in G^H. As aforementioned, $G_i \in G^H$ is a subgraph that contains a seed v_s and its adjacent set $\text{Adj}(v_s)$, $c_{i,j} = 1$ represent the edge connection between two nodes (v_i, v_j) and $e_{i,j}$ represent the edge weight. Hence, we extract the potential SCN using the following steps:

Step 1: (Initialization) set $i \leftarrow 1$

Step 2: while $i \leqslant \#(G^H)$ do

Step 3: (Simplification) given $G_i \in G^H$, set $c_{t,u} \leftarrow 1$ and $c_{t,r} \leftarrow 0$, where $v_t, v_u, v_r \in \text{Adj}(v_s)$, $v_u = \text{argmax}\{e_{t,u}\}$, and $v_r \neq v_u$; set $cn \leftarrow 0$

Step 4: (Optimization) given $v_k \in \text{Adj}(v_s)$, if $e_{s,t}$ and $e_{t,k}$ satisfy Eqs. (4)–(6), set $c_{s,t} \leftarrow 0$, $c_{t,k} \leftarrow 1$ and $cn \leftarrow cn + 1$

Table 1
Component of the ordered subgraphs after the hierarchical representation of TCN associated with the example weighted graph in Fig. 3.

Subgraph	Seed	Adjacent set	Subgraph	Seed	Adjacent set	Subgraph	Seed	Adjacent set
G_1	7	{5, 1, 6, 10, 9, 8}	G_5	10	{7, 9, 11}	G_9	4	{5, 3}
G_2	5	{7, 1, 6, 2, 4}	G_6	9	{7, 10, 8, 11}	G_{10}	11	{10, 9}
G_3	1	{7, 5, 6, 2}	G_7	8	{7, 6, 9}	G_{11}	3	{2, 4}
G_4	6	{7, 5, 1, 8}	G_8	2	{5, 1, 3}			

Step 5: repeat Step 4, until $cn = \#(\text{Adj}(v_s))$ or $d(v_s) = 2$; set $i \leftarrow i + 1$
Step 6: end while
Step 7: return one potential SCN

$$\frac{e_{t,k} - e_{s,t}}{e_{t,k}} > f, (f > 0) \tag{4}$$

$$0 < e_{s,t} < 0.5M \tag{5}$$

$$0.5M \leqslant e_{t,k} \leqslant M \tag{6}$$

where $M = \max\{e_{i,j} : i, j = 1, \cdots n\}$ indicates the maximum overlap between the image pairs in an UAV collection.

Steps (3–5) show the procedure of simplifying a subgraph. The simplification is performed under the assumption that the image corresponding to the seed $v_s \in V_i$ has the highest importance in 3D scene reconstruction as compared to its adjacent set $\text{Adj}(v_s) \in V_i$. Hence, we keep the edges between v_s and $\text{Adj}(v_s)$ as default in Step 3, and delete only the edges which are neither incident to v_s nor possess maximum weight of any node in $\text{Adj}(v_s)$. This step ensures each node $v_t \in \text{Adj}(v_s)$ is involved in at least a 3-tuple configuration, that is, $d(v_t) \geqslant 2$. It needs to be noted that although the edge connection between two nodes is deleted in processing the subgraph, the edge weight remains stored and is used for further optimization. In Step 4, the optimization is performed by testing whether the target factor f has been met based on our experimental experience. As a result, the previously deleted edges can be recovered through the optimization in Step 4. For instance, in case of $v_r = v_k$, the deleted edge connection $c_{t,r} \leftarrow 0$ in Step 3 can be recovered as $c_{t,r} \leftarrow 1$ after Step 4. In Step 5, one criterion that will stop the procedure in advance is that node degree $d(v_s) = 2$. This stopping criterion ensures the seed be involved in a 3-tuple configuration. Fig. 4 shows an illustrative example of simplifying subgraph G_1 with seed node {7} and its adjacent set {5, 1, 6, 10, 9, 8}. In particular, the use of Eqs. (4)–(6) during step 4 forces the algorithm to delete the edge connection as $c_{1,7} \leftarrow 0$ (although it has a higher priority) and recover $c_{1,6} \leftarrow 1$. Finally, out of 12 edges from the original subgraph G_1, a set of 4 edges {$c_{1,7}, c_{5,7}, c_{6,8}, c_{7,9}$} gets deleted.

These procedures are repeated, one subgraph at a time, until no remaining subgraph with redundant edges is existing, and finally producing a potential SCN. Fig. 5 shows the schematic representation of SCN extraction from the weighted TCN in Fig. 3 using the proposed HDB-MST algorithm. Fig. 5a indicates that the node set in the TCN is divided into four different layers. Fig. 5b shows the extracted potential SCN after processing all subgraphs from the hierarchically structured TCN in Fig. 5a. We call the whole procedure from one weighted TCN to the potential SCN as one

HDB-MST round (HMR). Subsequently, we update the weighted TCN as the potential SCN for processing another HMR. The HMR is repeated until all the edges in TCN stop changing, which results in the final SCN (Fig. 5b).

2.3. SCN embedded SfM

The SCN embedded SfM starts with the extraction of distinctive features from each image, using the Scale Invariant Feature Transform (SIFT) operator (Lowe, 2004) in our case. The SIFT operator, however, might fail to work on centimeter-resolution and richly textured UAV images, which have extremely large number of features. To improve the feature extraction performance, we adopt a divide-and-conquer strategy that divide one image into a set of sub-blocks, independently extract features and merge them into a complete file (Xu et al., 2014a).

Subsequently, tie-points are matched only in the image pairs with explicit connections within the SCN. To remove remaining matching errors, a RANSAC (Fischler and Bolles, 1981) estimation of the fundamental matrix is performed on each pair of matched images. We then generate a set of feature tracks, each of which records the position of a tie-point in different images. For increased speed and reliability, we delete the tracks which contain tie-points in less than three images.

Once the features are generated, we recover a set of camera parameters C_i and a 3D location X_k for each feature track through a joint minimization of a non-linear objective function $y(C, X)$. For F feature tracks $X = \{X_k\}_{k=1}^{F}$ in m cameras $C = \{C_i\}_{i=1}^{m}$, the minimization is achieved by means of a projection equation P that considers the accumulative projection errors of all the feature tracks as:

$$y(C, X) = \sum_{i=1}^{m} \sum_{k=1}^{F} w_{ik} \|q_{ik} - P(C_i, X_k)\|^2 \tag{7}$$

where q_{ik} indicates the measured position of the kth feature track in image i, $q_{ik} - P(C_i, X_k)$ indicates the projection error of kth feature track in image i. The variable $w_{ik} = 1$ if image i observes the kth feature track, otherwise $w_{ik} = 0$.

We use a sparse bundle adjustment (Lourakis and Argyros, 2004) to solve the minimization problem. Rather than estimating the parameters for all cameras and feature tracks at once, we add one camera at a time. We begin the estimation with the image pair that has the largest number of tie-points. Following that, the camera parameters and the 3D point positions are obtained in an arbitrary coordinate system, which is defined by the projection centers and baseline of the two initial images, if no additional information (e.g. ground control points) is available.
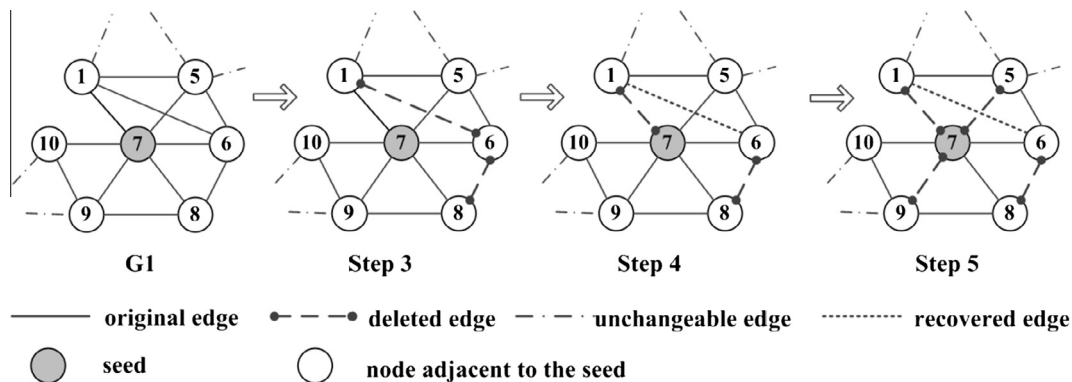

Fig. 4. Illustrations of simplifying subgraph G_1 with node set $V_1 = \{7, 5, 1, 6, 10, 9, 8\}$.
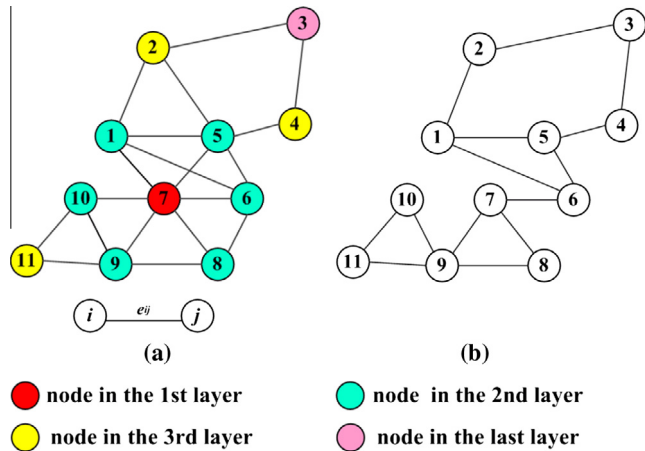
Fig. 5. An example of the schematic representation of SCN extraction from a weighted TCN using HDB-MST algorithm. (a) hierarchical TCN representation and (b) the final SCN that equals to the potential SCN after one HMR.

2.4. Method comparison and performance evaluation

We compare the proposed SCN-SfM method with three other SfM methods. Firstly, we use the traditional method, labelled as DN-SfM to process the UAV images (Snavely et al., 2006). Then, we compare our approach to the TCN-SfM method from Rupnik et al. (2013), which investigates TCN embedment for the SfM-based 3D scene reconstruction and is largely similar to the initial TCN computation as described here. Furthermore, we compare SCN-SfM to the MCN-SfM method from Alsadik et al. (2014), which constructs the matching graph from an initial SfM method based on resampled images.

We evaluate the performances of the proposed SCN-SfM method concerning three aspects: image matching, scene completeness, and geometric accuracy. To evaluate the performance of image matching, we concentrate on efficiency and feasibility, respectively. For efficiency assessment, we use the number of image pairs involved in matching as a proxy for run time, as proposed by Snavely et al. (2008). For feasibility assessment, we analyze the distribution of tie-points and quantify the number of correspondences between image pairs (Xu et al., 2014b). To evaluate the scene completeness, we use both qualitative and quantitative measures with regards to geometry coverage and to the number of retrieved 3D points. To evaluate the geometric accuracy, the point cloud is transformed from the arbitrary coordinate system to the user's coordinate system. There are two main solutions for the coordinate transformation. One follows the conventional photogrammetry principle that add at least three GCPs in the bundle adjustment to define the correct mapping datum (Rupnik et al., 2015). This solution can be achieved using commercial software e.g. Pix4Dmapper, PhotoScan, and open source software Micmac (http://logiciels.ign.fr/?-Micmac,3-). The other is achieved by identifying some GCPs in a densified point cloud and computing an appropriate 3D similarity transformation (Fonstad et al., 2013; Westoby et al., 2012). Residuals at GCPs are then used as a quality indicator. This solution can be achieved by following a 3D shape registration method based on the ICP algorithm (Besl and McKay, 1992). We are aware of the fact that the former one would most probably lead to better results, since block deformation, which is a result of remaining systematic errors in camera self-calibration, will be reduced. For practical applications such a method is highly suggested. In this study, however, we are interested only in comparing the methods relative to each other. Since the bundle block adjustment, including self-calibration method is the same for all cases to be compared, it seems even better to use the rigid trans-

Table 2
Theoretical comparison of image matching complexity in DN-SfM, MCN-SfM, TCN-SfM and the proposed SCN-SfM methods.

Method	DN-SfM	MCN-SfM	TCN-SfM	SCN-SfM
Computation	$\frac{n(n-1)}{2}$	$\frac{s(s-1)}{2}$	$un - \frac{u^2+u}{2}$	$vn - \frac{v^2+v}{2}$
Complexity	$O(n^2)$	$O(s^2)$	$O(n)$	$O(n)$

n is the number of images, s is the number of images remained in MCN, which is smaller than n. u and v are two constants, which are much smaller than n, and indicate the maximum degree of node in TCN and SCN, respectively.

formation, assuming that effects from different matching graphs are better visible. Then we compare the transformed coordinates of GCPs with GPS measurements for a relative ranking of the methods in the experimental analysis (Westoby et al., 2012).

Table 2 lists the computational complexities of the four methods in image matching. The complexities of DN-SfM and MCN-SfM are $O(n^2)$ and (s^2) $(n > s)$, respectively. The complexities of TCN-SfM and SCN-SfM are $O(n)$, since the two methods avoid traversal matching by taking into account the constraints of topological connectivity of image pairs. It should be stressed that the theoretical analysis is just an approximate one, and actual computation times may differ in practice. Experimental results can be found in Section 4.3.1.

We use the popular open source SfM system Bundler (Snavely, 2010) to obtain the sparse point clouds for the experiments. Bundler is written in C and C++ that allows us to embed the networks, in particular TCN and SCN, in the SfM method. The Patch-based Multi-View Stereo Software (PMVS-Version2) (Furukawa and Ponce, 2012) is used to densify the sparse point clouds. Notably, for comparison purposes, all the parameters in Bundler and PMVS2 are assigned the same values for the four methods.

3. Experimental test in three sites

In this study we conducted three experiments in Site A, B, and C in China to validate the proposed SCN-SfM method. Site A covers an area of ~3 × 7 km² centered at the campus of the China University of Mining and Technology (CUMT) in Xuzhou, Jiangsu Province and it includes mainly vegetation, roads, and built-up areas. Site B covers an area of ~200 × 400 m² at Gutian Village, Fujian Province and is characterized by dense housing facilities and lush vegetation (Xu et al., 2014b). Site C is a suburban landscape with an area of ~150 × 500 m² located at Linyi, Shandong Province. This site is relatively monotonous, with a couple of individual buildings.

Two fixed-wing UAVs were employed to capture images for Site A and C, while one octocopter UAV was employed to capture images for Site B. Each flight was automatically controlled by a

Table 3
Parameters for the three experimental sites.

| Parameter	Site A	Site B	Site C
UAV model	Fixed-wing	Octocopter	Fixed-wing
Flying height above ground (m)	450	100	120
Forward/lateral overlap (%)	80/60	80/60	80/60
Camera model	Canon EOS 5D mark II	Canon EOS 5D mark II	Panasonic DMC-FX75
Focal length (mm)	35	35	24
Number of images	947	45	56
Sensor size (mm × mm)	36 × 24	36 × 24	6.1 × 4.6
Image size (pixel × pixel)	5656 × 3744	5656 × 3744	4320 × 3240
GSD (cm)	12	2.6	1
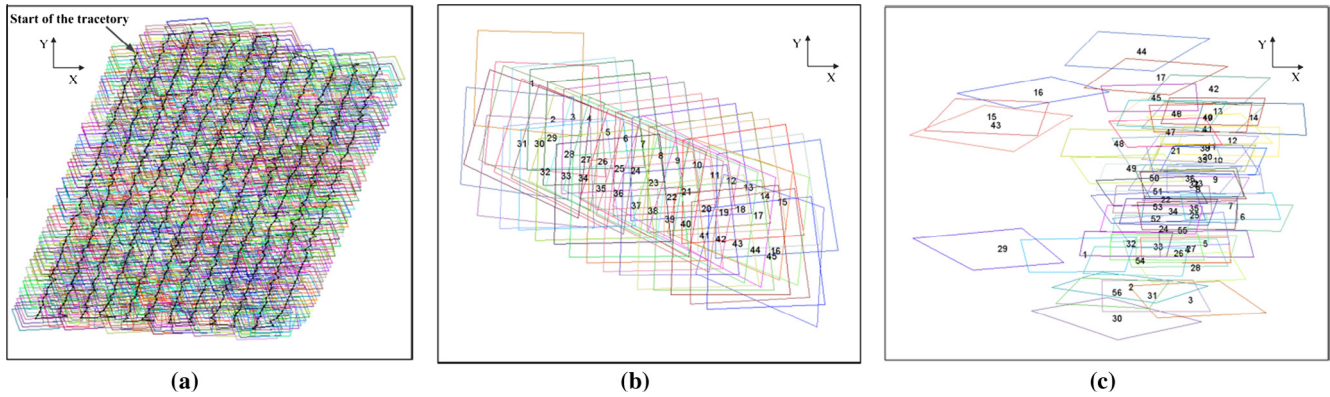
Fig. 6. Overview of UAV collections. (a–c) footprints of images in Site A, B, and C, respectively. The numbers in (b and c) represent the order of images captured by the flight control system.

ground station with a predefined trajectory. The flight control data of each flight were obtained by a low-cost flight control system. The geometrical resolution, indicated by ground sampling distance (GSD) varies from 1 cm to 12 cm. Table 3 lists the details of the parameters for the three experiments.

In addition, a total of 15 ground control points (GCPs) were well-distributed in Site B. These GCPs were fixed as cross markers with the edge length of 20 cm, which can clearly be identified in the UAV images. A GPS-RTK field survey was conducted before the flight, resulting in a nominal standard deviation below 2 cm for each GCP.

Fig. 6 gives an overview of UAV collections. Figs. 6(a–c) present the footprints of the images in Site A, B, and C, respectively. Figs. 6 (a and b) indicate two well-overlapped flights for Site A and B. Fig. 6c shows an irregular-overlapped flight, resulting from unfavorable weather conditions.

4. Experimental results and comparison

4.1. TCN construction

Fig. 7 shows the results of TCN construction for Site A, B, and C, respectively. It needs to be mentioned that from the 947 images on Site A, only a subset of 126 images (called as Site As) from four regular flight lines was selected for the following studies. Figs. 7(a–c) show the flight information and indicate the conjunctions between flight lines of Site As, B, and C, respectively. In these figures, the range of images from each flight line is shown in the main diagonal. The solid lines off the diagonal indicate the connected image pairs between different flight lines. The dotted line in Fig. 7a shows that no connected image pair can be found between two non-adjacent flight lines, e.g. line 1 and line 4. Figs. 7(d–f) show the adjacency matrixes of the corresponding TCNs of Site As, B, and C, respectively. We constructed a $n \times n$ grid for each site to visualize the distribution of connected image pairs and mapped their overlap onto the corresponding grids. For illustrative purpose, we visualized the overlap between the images of the three sites by the same range of (0–0.85). Fig. 7d shows that the overlap of one pair of neighboring images along the heading direction is larger (0.7–0.85) than that from two adjacent lines (0.3–0.45). This figure also shows that the image pairs with minimal overlap (e.g. overlap lower than 0.25) in the TCN account for a large partition of the connected image pairs. In addition, the overlap of an image pair from two non-adjacent flight lines is rather small, varying from 0.02 to 0.08 (Fig. 7d). The overlap between images in Fig. 7e exhibits a similar behavior as Fig. 7d but with larger overlapping values.

Nevertheless, the overlap behavior in Site C, where the images were collected in a more irregular flight (Fig. 7f) is different from those in Site As and B. Unlike the TCNs of Site As and B, very few image pairs have overlap larger than 0.7 and the majority have overlap 0.1–0.5.

4.2. SCN extraction

In Fig. 8 the extracted SCNs of Site As, B, and C from the corresponding TCNs by the proposed spanning tree, e.g. HDB-MST are shown. Figs. 8(a and b) respectively indicate the SCNs of Site As and B from two well-distributed flights. The SCNs shown in the two figures maintained almost all the pairs of neighboring images along the heading direction, producing the largest overlap. Compared to the corresponding TCNs in Figs. 7(d and e), the SCNs in Figs. 8(a and b) maintained much fewer connected image pairs from two adjacent flight lines. Fig. 8c shows the SCN of Site C from an irregular flight. This figure indicates that the distribution of the connected image pairs represented by the SCN of Site C is different from those of Site As and B. Specifically, for Site C many image pairs from non-adjacent flight lines are represented within the SCN, while for Site As and B no such image pairs are available. Besides, from Fig. 8c we can also observe that the connected pairs of neighboring images along the heading direction have smaller overlap than those from two adjacent or even non-adjacent flight lines represented by the SCN.

Fig. 9 plots the curves of the intermediate number of edges remaining in the weighted TCNs as a function of the HDB-MST rounds performed on Site As, B, and C. This figure indicates that the number of edges in the TCNs show an exponential decrease with the rise of HDB-MST rounds during SCN extraction. Moreover, the convergence rates of the curves for extracting SCNs were independent from the number of edges in the initial TCNs. For instance, the number of HDB-MST rounds for extracting the SCN of Site As is 6, corresponding to 1976 edges in its initial TCN (Fig. 9a); while the number of HDB-MST rounds is 7 for Site B, corresponding to 672 edges in its initial TCN (Fig. 9b). Notably, the convergence rate in Fig. 9c is faster than those in Figs. 9(a and b), with only 2 HDB-MST rounds to extract the SCN with 133 edges from its initial TCN with 310 edges. The number of edges in the SCN was determined by two parameters, e.g. the node degree $d(v_i)$ and the factor f used for deleting the edges of subgraph G_i in TCN. We set the parameters, for instance, $d(v_i) = 2$ and $f = 1$ to make sure each node in the filtered subgraph was involved in at least a 3-tuple configuration and the subgraph contained the minimally required edges.
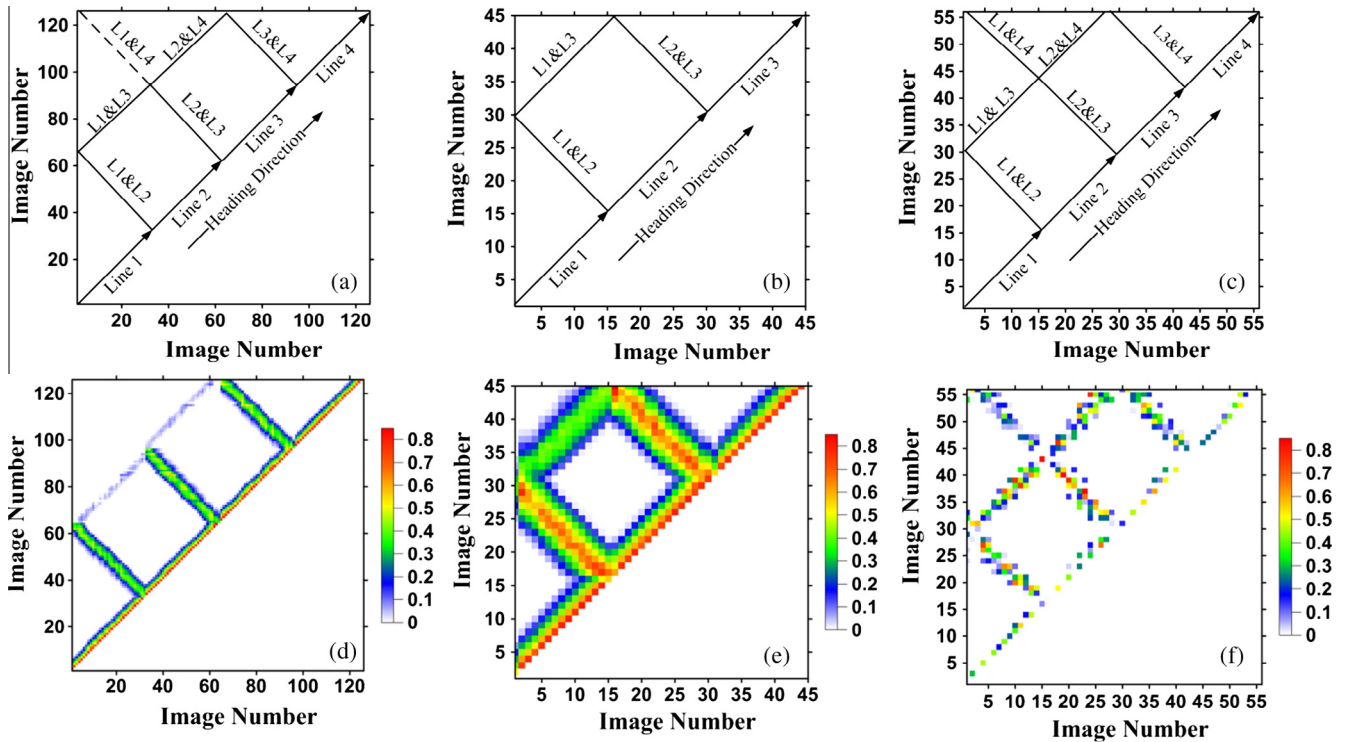
Fig. 7. Results of TCN construction. (a–c) indicate the schematic maps of the flights for Site As, B, and C, respectively. Both Site As and C contain four flight lines, whereas Site B contains three flight lines. The connected image pairs along the heading direction are displayed in diagonal, while those from two flight lines (L*i*&L*j*) are displayed off the diagonal. (d–f) show the TCNs of Site As, B, and C, respectively. The color bars in (d–f) indicate the range of overlap (0–0.85) between image pairs.
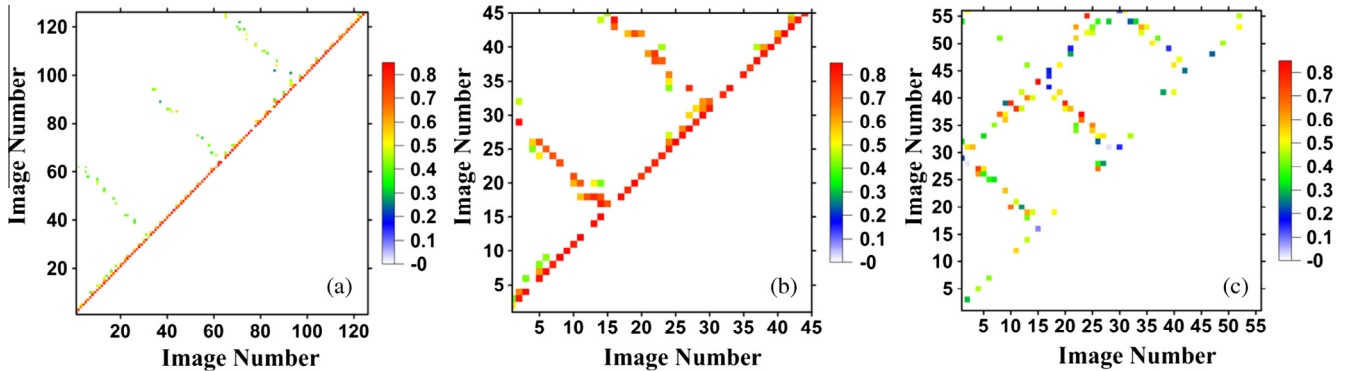

Fig. 8. Results of SCN extraction. (a–c) indicate the SCNs of Site As, B, and C, respectively. The color bars indicate the range of overlap (0–0.85) between the image pairs in the three sites. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4.3. Method comparison and performance evaluation

For this section we compared the proposed SCN-SfM method with the aforementioned three methods DN-SfM, MCN-SfM, and TCN-SfM. The comparison was focused on three aspects in terms of image matching efficiency, scene completeness, and geometric accuracy.

4.3.1. Image matching efficiency

We used the number of edges in the network, corresponding to the image pairs involved in matching, as a proxy for run time. Fig. 10 displays the number of edges in the networks of DN, MCN, TCN and the proposed SCN of Site As, B, and C. This figure shows that – as expected – the DNs have the largest number of edges with value 7875, 990, and 1540 for Site As, B, and C, respectively. On the contrary, the SCNs of the corresponding three sites

have the smallest number of edges, with value 234, 93, and 133, respectively. Furthermore, the number of edges in MCNs is close to that of TCNs for Site As and B, but is much larger (1081) than that (310) of TCN on Site C.

Following the numbers shown in Fig. 10, we fitted curves of the number of edges in DNs, MCNs, TCNs and SCNs of the three sites as a function of the number of images (Fig. 11). The horizontal axis shows the number of images of Site B, C, and As in order with 45, 56, and 126, respectively. It indicates that the number of edges in DNs has a quadratic increase with the number of images, while the number of edges in MCNs, TCNs and SCNs exhibit a linear increase, where the increase rate in SCNs is the least. The sub-curve in Fig. 11 shows the fitted curve by the number of edges in MCNs of Site B, C, and As with the remained image size 35, 47, and 68, respectively. A quadratic increase was given in this sub-curve. This was confirmed in Table 2. In general, the number of
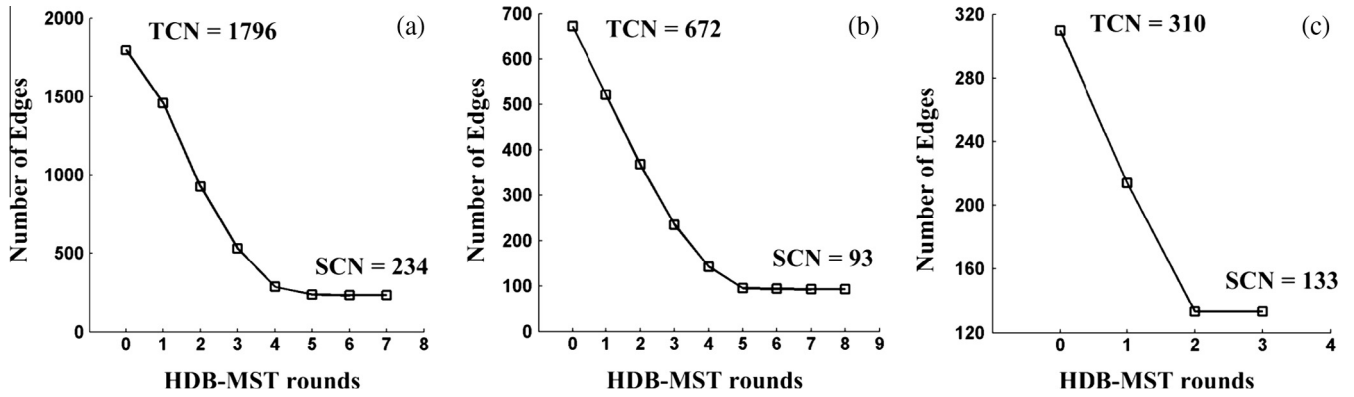
Fig. 9. The number of edges in the intermediate networks of SCN extraction at each iteration for (a) Site As, (b) Site B, and (c) Site C.
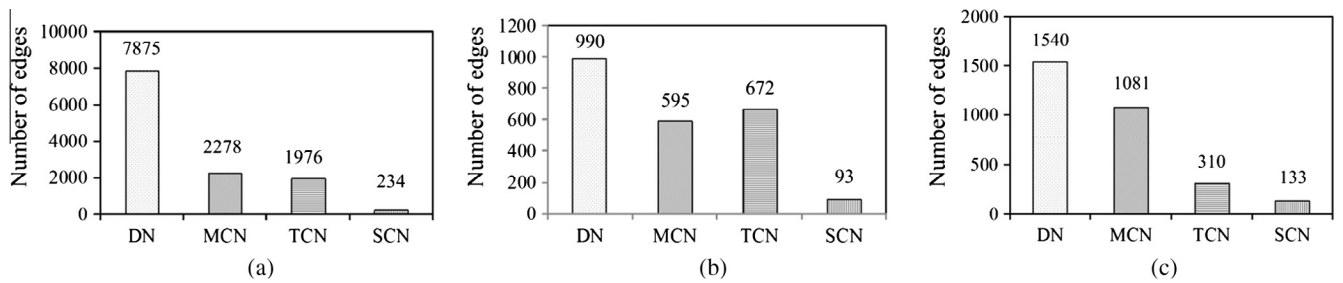

Fig. 10. Comparison of the number of edges in the networks of DN, MCN, TCN, and SCN for (a) Site As, (b) Site B, and (c) Site C.
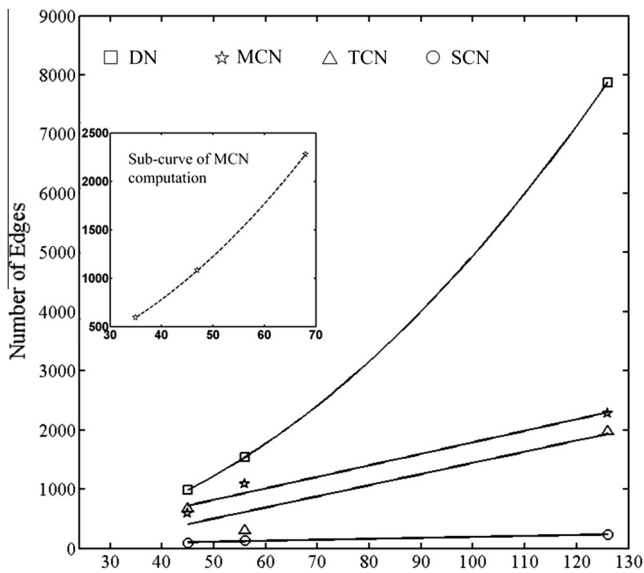

Fig. 11. The number of edges and fitted curves in the networks of DN, MCN, TCN, and SCN on Site As, B, and C. The sub-curve is fitted by the number of edges in MCN with the remained images on Site As, B, and C.

images within MCN-SfM is much smaller because not the connections between several image pairs get removed, but entire images are excluded from the block.

Fig. 12 illustrates the image connections of Site As in DN, MCN, TCN, and SCN in the object space domain. The images acquired in four parallel flight lines, were projected onto the user's coordinate system using the flight control data, and the edges in the networks represent the connections between the images. It needs to be noted that all the 126 images of Site As were involved in DN,

TCN, and SCN, while only 68 images were remained in MCN. In addition, both DN and MCN employed a one-by-one image matching strategy, making the networks much denser (Figs. 12a and b) than TCN and SCN (Figs. 12c and d) where only the topologically connected edges were maintained. Hence, the connectivity of SCN was much sparser than that of TCN.

In Fig. 13 the distribution of tie-points among the image pairs of Site As, yielded using DN-SfM, MCN-SfM, TCN-SfM, and SCN-SfM methods is displayed. We constructed a 126×126 grid for visualizing the distribution of tie-points and mapped the number of tie-points between an image pair onto the corresponding grid. The number of tie-points on the grid was proportionally visualized using a color scheme. Fig. 13a shows the distribution of tie-points from the DN-SfM method. It indicates some "noise" by low color intensity between image pairs without connection. The "noise" indicates the mismatches, which were caused by the threshold technique (Lowe, 2004; Snavely et al., 2006). Fig. 13b shows the distribution of tie-points from the MCN-SfM approach. Compared to DN-SfM, the MCN-SfM method reduced the number of images, which would neither be involved in matching nor in the scene reconstruction. The removal of images from the graph resulted in the strip-like pattern visible in Fig. 13b. At the same time the mismatches caused by spatially non-connected image pairs remained. Moreover, many image pairs that have numerous tie-points were removed from the MCN, while some image pairs with small amount of tie-points remained. Although TCN-SfM reduced the mismatches between images without connection, it maintained many image pairs having very few tie-points, which were represented by darker color cells in Fig. 13c. The distribution of tie-points between images obtained by SCN-SfM is depicted in Fig. 13d. This figure shows that only a small set of edges corresponding to the image pairs with the largest number of tie-points were involved in matching. Furthermore, it is obvious that the distribution of tie-points is consistent with the distribution of the edges in Fig. 8a.
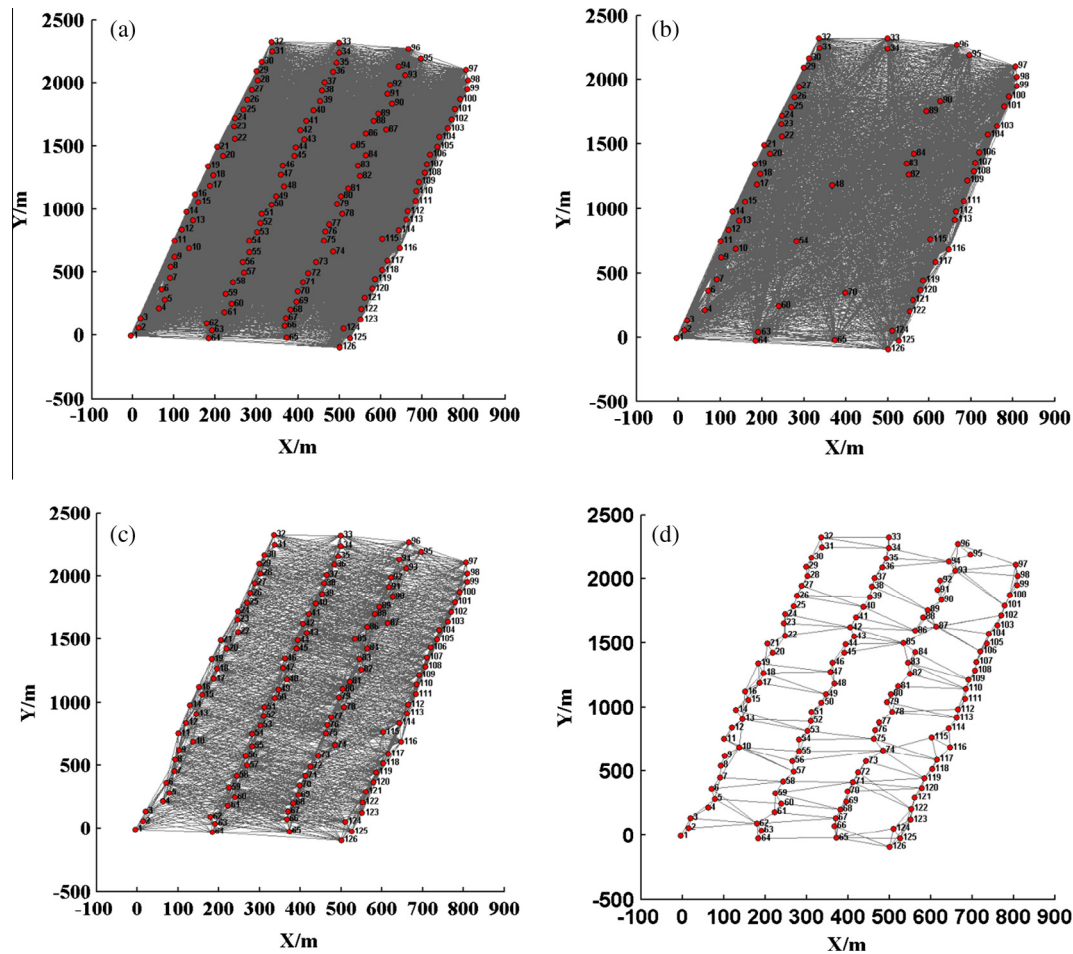
Fig. 12. Visualization of network (a) DN, (b) MCN, (c) TCN, and (d) SCN from the 126 images of Site As. The positions of UAV images are shown in the user's coordinate system after a 2D projection from the flight control data. Each image is represented by a node in the network, where the edge between nodes indicates their connection within the matching graph.

4.3.2. Geometric completeness

Fig. 14 shows the 3D scene geometries of Site As with 126 images recovered by DN-SfM, MCN-SfM, TCN-SfM, and SCN-SfM methods. Note that SCN-SfM ended up with a comparable scene geometry (Fig. 14d) to those of DN-SfM (Fig. 14a) and TCN-SfM (Fig. 14c). For MCN-SfM, however, it resulted in a long gap in the middle of the scene geometry, where several coverage images were deleted (Fig. 14b). The 3D scene geometry for Site B and C were obtained in a similar manner, respectively. The results indicated no significant difference between the completeness of scene geometry resulted from the four methods on either Site B or Site C.

Table 4 indicates the number of retrieved 3D points on Site As, B, and C using the four methods. The results reveal that SCN-SfM yields an equivalent number of points as DN-SfM and TCN-SfM. As compared to MCN-SfM, the proposed SCN-SfM produces nearly twice the amount of points on Site As. Moreover, the scene geometry resulting from SCN-SfM on Site A with 947 images was depicted in appendix as Fig. A1. Although only 745 of the 947 images were involved in the final 3D scene reconstruction by using the SCN-SfM method, the major partition of the scene geometry had been successfully recovered. The part of scene geometry which failed to be recovered was entirely covered by vegetation with textureless features.

4.3.3. Geometric accuracy

Due to the lack of ground control points (GCPs) in Site A and C, we used only the dataset of Site B to quantify the geometric accu-

racy of the proposed SCN-SfM method. Fig. 15 shows the dense point cloud of Site B using SCN-SfM followed by the PMVS method. From a pure visual check the other three dense point clouds of Site B based on DN-SfM, MCN-SfM, and TCN-SfM methods show no variation to the SCN-SfM-derived result. The densities of these point clouds were about 40 points/m², which allowed us to accurately select the GCPs.

All the 15 GCPs in Site B were accurately selected from the dense point clouds and used for accuracy assessment. Here, a 7-parameter based similarity estimation method was used for the coordinate system transformation, resulting in a set of residuals associated with *x*-, *y*-, *z*-components for each GCP.

Fig. 16 shows the residual vectors at GCPs resulting from the four network-based SfM methods. Since a similarity transform was performed, the shape of model did not get modified. In fact, only systematic shifts and rotations got averaged at the GCPs. This means that the residuals somewhat represent remaining tension and block deformation and we could perform at least some relative analysis. The results for different methods are shown through residual diagrams in Fig. 17 as well. The RMSE of all the GCPs at *x*-, *y*-, and *z*-components are given, respectively. Compared to the nominal GSD value which is 2.6 cm for this dataset, the RMSE is quite high – up to 4 times of GSD value for *x/y* and 13 times of GSD value for *z*. We also observe quite large, partly opposite *z*-residuals, which indicate some block deformation. In case of MCN-SfM method it seems to be the largest. Since block deformation might be a result of insufficient sensor calibration, we can con-
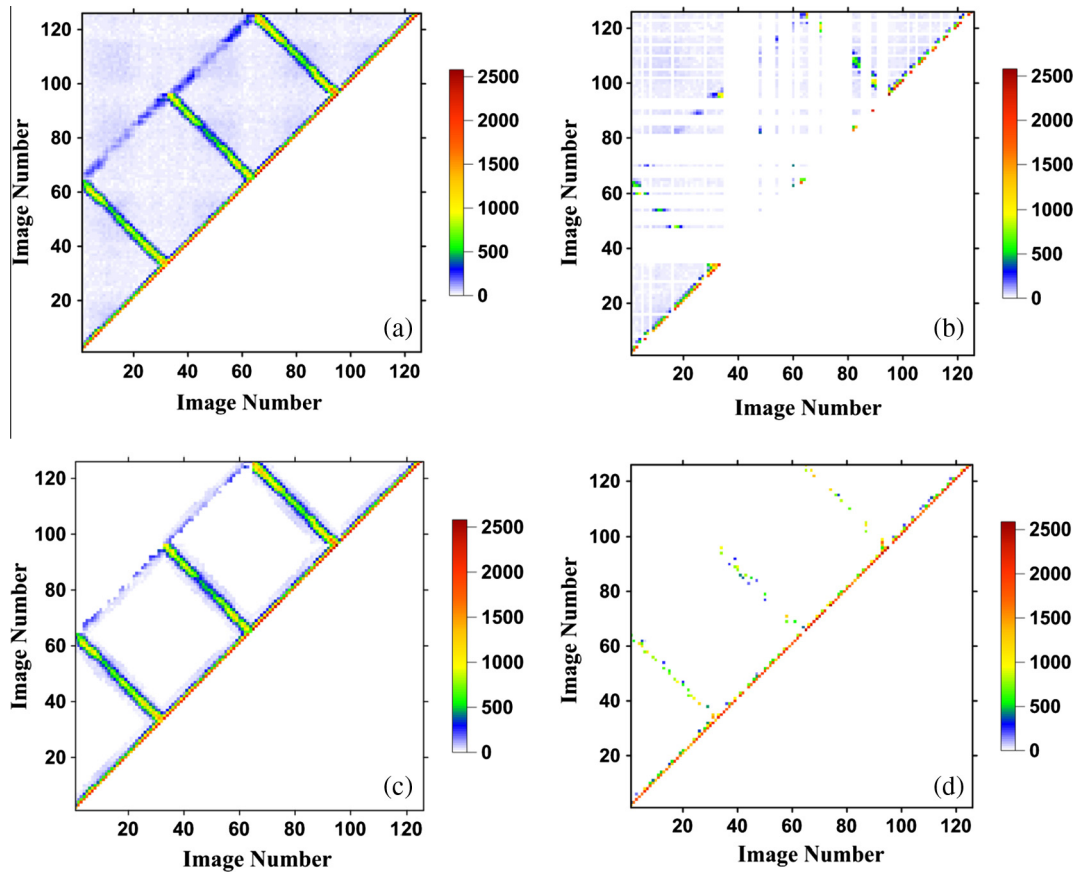
Fig. 13. Tie-points between the image pairs from method (a) DN-SfM, (b) MCN-SfM, (c) TCN-SfM, and (d) SCN-SfM. The color bars on the right side indicate the number of tie-points between the image pairs.
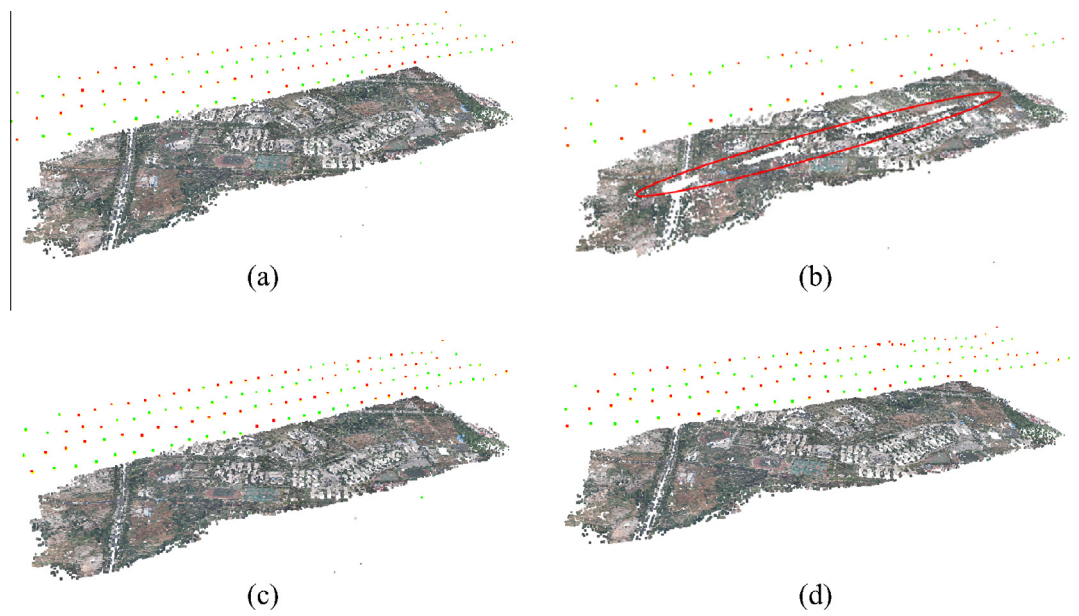

Fig. 14. Reconstructed 3D scene geometry from 126 UAV images on Site As using method (a) DN-SfM, (b) MCN-SfM, (c) TCN-SfM, and (d) SCN-SfM. Besides the point cloud of the scene, the positions of cameras are also recovered and represented by red or green points above the scene geometry in adjacent flying lines. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 4
Comparison of the number of points reconstructed on Site As, B, and C using the four methods.

Site	Method			
	DN-SfM	MCN-SfM	TCN-SfM	SCN-SfM
As	104,448	53,698	101,781	101,210
B	30,308	24,808	30,429	30,415
C	26,704	23,470	24,035	23,628

clude that the network resulting from MCN-SfM method is the least suitable among our examples.

5. Discussion

This study investigated the embedment of SCN in SfM for efficient 3D scene reconstruction from large quantity of UAV images. Our results confirm that SCN embedment largely contributes to computational savings. Many studies on computational savings of SfM method for 3D scene reconstruction can be found in the literature (Alsadik et al., 2013; Rupnik et al., 2013; Snavely et al., 2008), particularly the recent work by Xu et al. (2013), who first investigated the SCN embedded SfM, where a minimum spanning tree was used for SCN extraction. Our method differs to the latter in some respects: First, the proposed method applies a variant of a maximum spanning tree for SCN extraction by using (a) importance weight based on both overlap and node degree, (b) hierarchical filter strategy, (c) degree bounded considerations, and (d) further optimizations; Second, a thorough experimental evaluation was provided with the proposed SCN-SfM method.

From the examples in this study it is clear that the number of edges in SCN, used to quantify the matching efficiency, is much less than those resulting from the compared networks. In this study, the SCN extraction relies on the bounded degree of the node together with the accumulative weight to others. In practice, the minimum node degree is recommended bounding to 2 to keep the SCN in at least a 3-tuple configuration with a minimally required number of edges. In addition, the extraction of SCN is independent of the SfM pipeline, because only the flight control data and camera hardware information are used as input. The computation time for SCN extraction is within a minute, even for hundreds of images. Furthermore, the proposed SCN-SfM is parallelizable with GPU techniques, as introduced by Xu et al. (2014a), and used in this study.

Our results also confirm that the reduction of the number of image pairs to be matched based on the SCN involves no loss in completeness and accuracy of the final scene geometry. Although the accuracy obtained by the SCN-SfM method was comparable to the one obtained by three other methods, further improvements are possible. For instance, the use of GCPs for bundle adjustment is advised for (Rupnik et al., 2015).

One extension of SCN extraction is to cope with the computational cost in bundle adjustment, as introduced by Toldo et al. (2015). This approach partitions a large image set into a couple of small sets based on the image degree and the image connectivity with base-line configuration. Our method, however, can be used to provide a more straightforward strategy to achieve that. Other alternative methods, for instance, the removal of redundant tie-points in the bundle adjustment also contribute to the computational savings (Mayer, 2014; Schaffalitzky and Zisserman, 2002), which can also be integrated into SCN-SfM method.

The results from Site A showed that textureless areas failed to be reconstructed (Fig. A1). This is due to an insufficient number of tie-points among image pairs. Several solutions on the redefinition of rules associated with feature extraction and matching would be potentially useful to solve this problem (Furukawa et al., 2004; Kawanishi et al., 2013; Saponaro et al., 2014). In this study, we used only the flight control data to generate the TCN. Further research should include the use of a terrain elevation model to generate a TCN for datasets in mountainous area with extensive elevation changes. Instead of relying on terrain models from third parties some approximate height models could be derived after the first SCN-based optimization, possibly on downsampled images: very often the tie-point clouds are good enough to compute some surface models, which can then be used in a new SCN initialization with better overlap information. The other problem mentioned above, regarding poorly textured areas, can then also be addressed by using tie-point statistics from the initial

Fig. 15. Dense point cloud with RGB color information of Site B reconstructed by the proposed SCN-SfM method (after densification using PMVS).
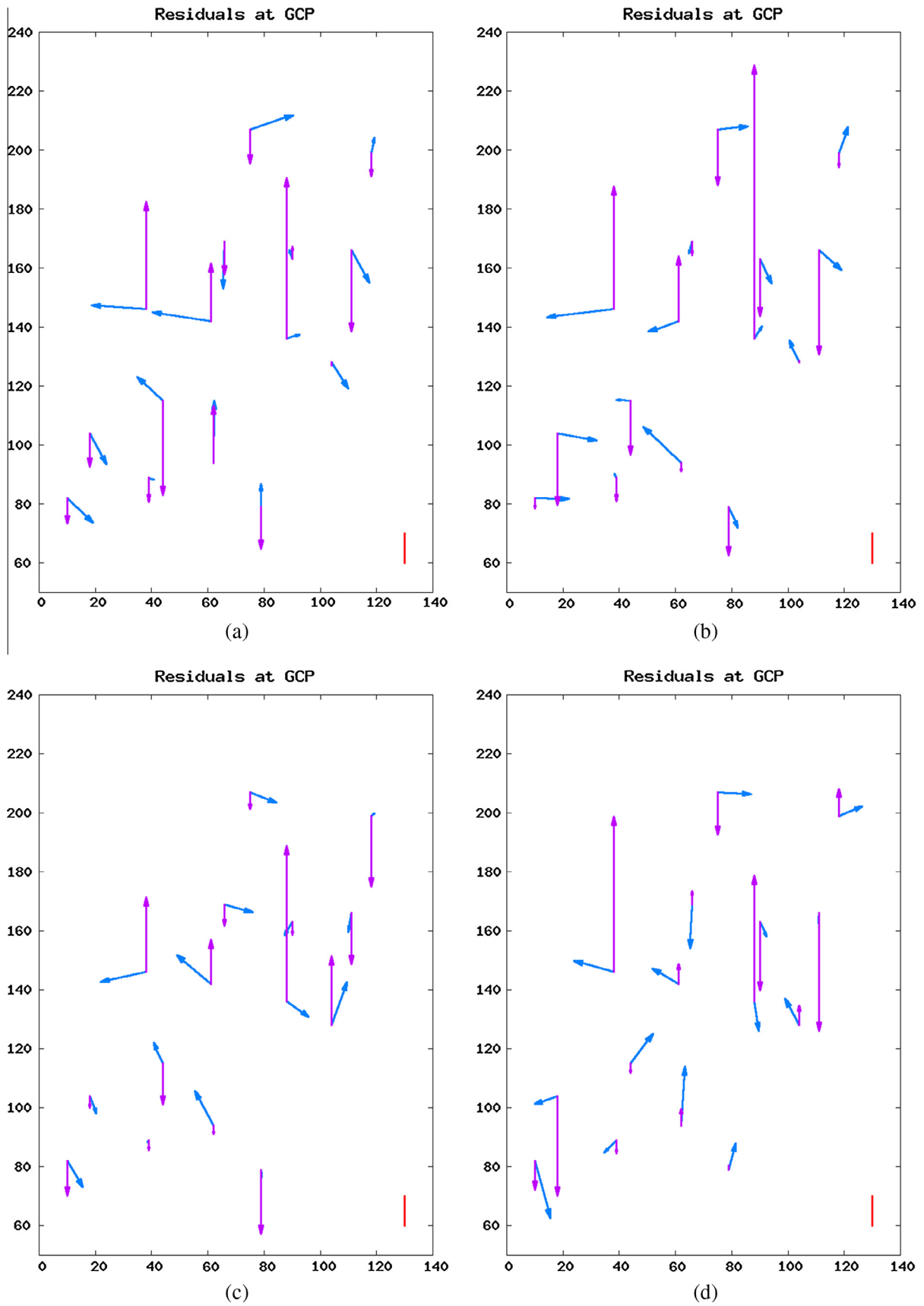
Fig. 16. Discrepancy vectors at GCPs resulting from method (a) DN-SfM, (b) MCN-SfM, (c) TCN-SfM, and (d) SCN-SfM. The red line is the scale bar that equals to 10 cm. The purple line represents the residual at *z*-component, and the blue line represents the residual at *xy*-component. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
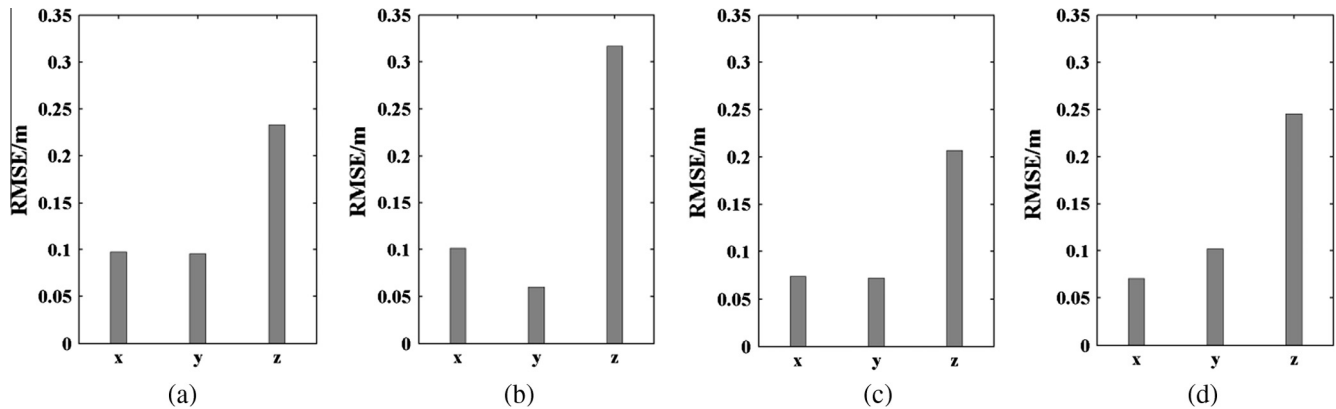
Fig. 17. Residuals (RMSE) at GCPs in x-, y-, and z-components in the 3D point clouds from method (a) DN-SfM, (b) MCN-SfM, (c) TCN-SfM, and (d) SCN-SfM.
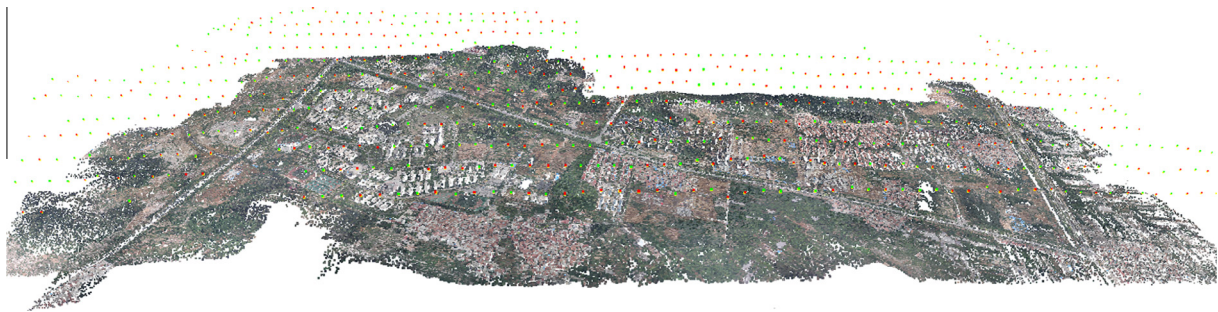

Fig. A1. Reconstructed 3D scene geometry on dataset A with a total of 947 UAV images captured by a fixed-wing UAV, where only 745 images were involved in the final reconstruction procedure.

matching step: edge weights might be decreased or increased according to tie-point matching performance.

6. Conclusions

SfM-based methods have already demonstrated their capabilities for 3D scene reconstruction from overlapping images in many applications. Nevertheless, it is still needed to overcome the major limitation in explosive increment of computation for large amount images. This paper investigated the use of SCN for computational savings of SfM in an embedded design to process large UAV image blocks. The attention was mainly devoted to image matching, one of the most time-consuming phases within SfM methods. Based on the flight control data, the footprints of a UAV collection can be effectively obtained for constructing a TCN, which identifies the connections between images by a weighted graph. We proposed a variant of the maximum spanning tree approach for extracting a SCN from the weighted TCN, where the use of hierarchical representation and node degree bounded strategies were investigated. We embedded the SCN into the SfM and proposed a novel SCN-SfM method. Experimental tests were done in three sites, and the results showed that the proposed SCN-SfM method outperformed the three other existing methods. We concluded that the SCN embedment significantly contributed to computational savings of image matching in SfM, and preserved the quality of 3D scene geometry in particular regarding completeness and accuracy.

Acknowledgement

We would like to thank the anonymous reviewers and the associate editor for their comments. We would also like to thank

Appendix A

See Fig. A1.

References

Alsadik, B., Gerke, M., Vosselman, G., 2013. Automated camera network design for 3D modeling of cultural heritage objects. J. Cult. Heritage 14 (6), 515–526.
Alsadik, B., Gerke, M., Vosselman, G., 2015. Efficient use of video for 3D modelling of cultural heritage objects. ISPRS Ann. Photogramm. Rem. Sens. Spatial Inform. Sci. 1, 1–8.
Alsadik, B., Gerke, M., Vosselman, G., Daham, A., Jasim, L., 2014. Minimal camera networks for 3D image based modeling of cultural heritage objects. Sensors 14 (4), 5785–5804.
Besl, P.J., McKay, N.D., 1992. Method for registration of 3-D shapes. Proc. Robotics-DL Tentative, 586–606.
Bulatov, D., Häufel, G., Meidow, J., Pohl, M., Solbrig, P., Wernerus, P., 2014. Context-based automatic reconstruction and texturing of 3D urban terrain for quick-response tasks. ISPRS J. Photogramm. Rem. Sens. 93, 157–170.
Castillo, C., James, M.R., Redel-Macías, M.D., Pérez, R., Gómez, J.A., 2015. SF3M software: 3-D photo-reconstruction for non-expert users and its application to a gully network. Soil 1 (2), 583–594.
Colomina, I., Molina, P., 2014. Unmanned aerial systems for photogrammetry and remote sensing: a review. ISPRS J. Photogramm. Rem. Sens. 92, 79–97.

Cover, T.M., Hart, P.E., 1967. Nearest neighbor pattern classification. Inform. Theor. IEEE Trans. 13 (1), 21–27.

Dandois, J.P., Ellis, E.C., 2013. High spatial resolution three-dimensional mapping of vegetation spectral dynamics using computer vision. Remote Sens. Environ. 136, 259–276.

Douterloigne, K., Gautama, S., Philips, W., 2010. On the accuracy of 3D landscapes from UAV image data. In: Proc. IEEE Int. Geosci. Remote Sens. Sympos. Honolulu, HI, 25–30 July, pp. 589–592.

El-Hakim, S.F., Beraldin, J.A., Picard, M., Godin, G., 2004. Detailed 3D reconstruction of large-scale heritage sites with integrated techniques. IEEE Comput. Graphics Appl. 24 (3), 21–29.

Feng, J., Wu, L., Ma, S., 2014. Research of 3D virtual scene generation and visualization based on images. In: Advances in Intelligent Systems and Computing. Springer, International Publishing, pp. 375–386.

Ferworn, A., Tran, J., Ufkes, A., D'Souza, A., 2011. Initial experiments on 3D modeling of complex disaster environments using unmanned aerial vehicles. In: Proc. IEEE International Symposium on Safety, Security, and Rescue Robotics, Kyoto, 1–5 November, pp. 167–171.

Fischler, M.A., Bolles, R.C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM 24 (6), 381–395.

Fonstad, M.A., Dietrich, J.T., Courville, B.C., Jensen, J.L., Carbonneau, P.E., 2013. Topographic structure from motion: a new development in photogrammetric measurement. Earth Surf. Proc. Land. 38 (4), 421–430.

Furukawa, Y., Ponce, J., 2012. Patch-based Multi-view Stereo Software (PMVS-Version 2) <http://francemapping.free.fr/Portfolio/Prog3D/PMVS2.html> (Accessed 2016 August 23).

Furukawa, Y., Sethi, A., Ponce, J., Kriegman, D., 2004. Structure and motion from images of smooth textureless objects. In: Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, pp. 287–298.

Gavril, F., 1987. Generating the maximum spanning trees of a weighted graph. J. Algor. 8 (4), 592–597.

Gerke, M., Kerle, N., 2011. Automatic structural seismic damage assessment with airborne oblique Pictometry© imagery. Photogramm. Eng. Rem. Sens. 77 (9), 885–898.

Gouveia, L., Moura, P., Ruthmair, M., Sousa, A., 2014. Spanning trees with variable degree bounds. Eur. J. Oper. Res. 239 (3), 830–841.

Held, M., Karp, R.M., 1970. The traveling-salesman problem and minimum spanning trees. Oper. Res. 18 (6), 1138–1162.

Helmi, B.H., Rahmani, A.T., 2014. Maximum spanning tree based linkage learner. AI Commun. 27 (3), 263–274.

Katagiri, H., Hayashida, T., Nishizaki, I., Guo, Q., 2012. A hybrid algorithm based on tabu search and ant colony optimization for k-minimum spanning tree problems. Expert Syst. Appl. 39 (5), 5681–5686.

Kawanishi, R., Yamashita, A., Kaneko, T., Asama, H., 2013. Parallel line-based structure from motion by using omnidirectional camera in textureless scene. Adv. Robot. 27 (1), 19–32.

Kong, X., Everett, H., Toussaint, G., 1990. The Graham scan triangulates simple polygons. Pattern Recogn. Lett. 11 (11), 713–716.

Koutsoudis, A., Vidmar, B., Ioannakis, G., Arnaoutoglou, F., Pavlidis, G., Chamzas, C., 2014. Multi-image 3D reconstruction data evaluation. J. Cult. Heritage 15 (1), 73–79.

Lourakis, M., Argyros, A., 2004. The design and implementation of a generic sparse bundle adjustment software package based on the levenberg-marquardt algorithm. Technical Report 340, Institute of Computer Science-FORTH, Heraklion, Crete, Greece.

Lowe, D.G., 2004. Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision 60 (2), 91–110.

Lucieer, A., Turner, D., King, D.H., Robinson, S.A., 2014. Using an Unmanned Aerial Vehicle (UAV) to capture micro-topography of Antarctic moss beds. Int. J. Appl. Earth Obs. Geoinf. 27, 53–62.

Mayer, H., 2014. Efficient hierarchical triplet merging for camera pose estimation. In: Lecture Notes in Computer Science. Springer, International Publishing, pp. 399–409.

Niethammer, U., James, M., Rothmund, S., Travelletti, J., Joswig, M., 2012. UAV-based remote sensing of the Super-Sauze landslide: evaluation and results. Eng. Geol. 128, 2–11.

PhotoScan, A., 2013. User Manual: Professional Edition, Version 0.9.1, AgiSoft LLC: Petersburg, Russia.

Pollefeys, M., Koch, R., Van Gool, L., 1999. Self-calibration and metric reconstruction inspite of varying and unknown intrinsic camera parameters. Int. J. Comput. Vision 32 (1), 7–25.

Rupnik, E., Nex, F., Remondino, F., 2013. Automatic orientation of large blocks of oblique images. ISPRS-Int. Arch. Photogramm., Remote Sens. Spatial Inform. Sci. 1 (1), 299–304.

Rupnik, E., Nex, F., Toschi, I., Remondino, F., 2015. Aerial multi-camera systems: accuracy and block triangulation issues. ISPRS J. Photogramm. Remote Sens. 101, 233–246.

Saponaro, P., Sorensen, S., Rhein, S., Mahoney, A.R., Kambhamettu, C., 2014. Reconstruction of textureless regions using structure from motion and image-based interpolation. In: Proc. IEEE International Conference on Image Processing, Paris, 27–30 October, pp. 1847–1851.

Scaramuzza, D., Martinelli, A., Siegwart, R., 2006. A flexible technique for accurate omnidirectional camera calibration and structure from motion. In: Proc. IEEE International Conference on Computer Vision Systems, Manhattan, New York, 4–7 January, pp. 45–45.

Schaffalitzky, F., Zisserman, A., 2002. Multi-view matching for unordered image sets, or "How do I organize my holiday snaps?". In: Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, pp. 414–431.

Snavely, N., 2010. Bundler: structure from motion (SFM) for unordered image collections <http://www.cs.cornell.edu/~snavely/bundler/>. (Accessed 2016 August 23).

Snavely, N., Seitz, S.M., Szeliski, R., 2006. Photo tourism: exploring photo collections in 3D. In: Proc. ACM transactions on graphics, Boston, Massachusetts, 30 July–3 August, pp. 835–846.

Snavely, N., Seitz, S.M., Szeliski, R., 2008. Skeletal graphs for efficient structure from motion. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, 23–28 June, pp. 1–8.

Snavely, N., Simon, I., Goesele, M., Szeliski, R., Seitz, S.M., 2010. Scene reconstruction and visualization from community photo collections. Proc. IEEE 98 (8), 1370–1390.

Sturm, P., Triggs, B., 1996. A factorization based algorithm for multi-image projective structure and motion. In: Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, pp. 709–720.

Toldo, R., Gherardi, R., Farenzena, M., Fusiello, A., 2015. Hierarchical structure-and-motion recovery from uncalibrated images. Comput. Vis. Image Underst. 140, 127–143.

Verdie, Y., Lafarge, F., Alliez, P., 2015. LOD generation for urban scenes. Assoc. Comput. Mach. 34 (3), 15.

Vetrivel, A., Gerke, M., Kerle, N., Vosselman, G., 2015. Identification of damage in buildings based on gaps in 3D point clouds from very high resolution oblique airborne images. ISPRS J. Photogramm. Remote Sens. 105, 61–78.

Westoby, M., Brasington, J., Glasser, N., Hambrey, M., Reynolds, J., 2012. 'Structure-from-Motion'photogrammetry: a low-cost, effective tool for geoscience applications. Geomorphology 179, 300–314.

Woodget, A., Carbonneau, P., Visser, F., Maddock, I., 2015. Quantifying submerged fluvial topography using hyperspatial resolution UAS imagery and structure from motion photogrammetry. Earth Surf. Proc. Land. 40 (1), 47–64.

Xu, Z., Wu, L., Chen, S., Wang, R., Li, F., Wang, Q., 2014a. Extraction of image topological graph for recovering the scene geometry from UAV collections. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 319–323.

Xu, Z., Wu, L., Liu, J., Shen, Y., Li, F., Wang, R., 2015. Modification of SfM algorithm referring to image topology and its Application in 3-Dimension reconstruction of disaster area. Geomatics Inform. Sci. Wuhan Univ. 40 (5), 599–606.

Xu, Z., Wu, L., Shen, Y., Li, F., Wang, Q., Wang, R., 2014b. Tridimensional reconstruction applied to cultural heritage with the use of camera-equipped UAV and terrestrial laser scanner. Rem. Sens. 6 (11), 10413–10434.

Xu, Z., Wu, L., Wang, Z., Wang, R., Li, Z., Li, F., 2013. Matching UAV images with image topology skeleton. In: Proc. IEEE International Geoscience and Remote Sensing Symposium Melbourne, VIC, 21–26 July, pp. 546–549.

Zhang, C., Kovacs, J.M., 2012. The application of small unmanned aerial systems for precision agriculture: a review. Precision Agric. 13 (6), 693–712.