VIEW GRAPH CONSTRUCTION FOR LARGE-SCALE UAV IMAGES: AN EVALUATION OF STATE-OF-THE-ART METHODS

Junhuan Liu [1,2], Yichen Ma [1], San Jiang [1,2,3,*], Qingquan Li [2], Wanshou Jiang [4], Lizhe Wang[1,3]

[1] School of Computer Science, China University of Geosciences, Wuhan 430074, China – jiangsan@cug.edu.cn
[2] Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ), Guangdong Shenzhen 518060, China
[3] Hubei Key Laboratory of Intelligent Geo-Information Processing, China University of Geosciences, Wuhan 430078, China
[4] State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430072, China

Commission II, WG II/1

KEY WORDS: Photogrammetry, Unmanned Aerial Vehicle, Structure from Motion, View Graph, Image Retrieval

ABSTRACT:

Structure from Motion (SfM) is a 3D reconstruction framework that has achieved great success on large-scale Unmanned Aerial Vehicle (UAV) images. Due to the high time consumption of feature matching, a matching candidate subset is obtained by image retrieval to improve efficiency. Bag of Word (BoW) based image retrieval has been widely used in SfM systems, but the large number of local features and the high dimension of the BoW vector cause the retrieval method time-consuming. Vector of Locally Aggregated Descriptors (VLAD) and learning-based NetVLAD perform well in image retrieval, and these vector representation methods are evaluated in this study. After images are transformed into vectors, Nearest Neighbour (NN) searching methods like Brute-force and KD-Tree are used to find similar images. But as the number of images and the vector dimension increase, Approximate Nearest Neighbour (ANN) searching methods like Hierarchical Navigable Small World (HNSW) and Locality-Sensitive Hashing (LSH) are considered to replace NN searching to avoid efficiency degradation. These vector searching methods are also evaluated in this study. The test results demonstrate that the optimal method VLAD with HNSW can speed up about 100 times in finding matching candidate subset. A view graph that guides scene partition and sub-scene reconstruction in parallel SfM can be created by the optimal method. With this view graph construction method, the efficiency of SfM is significantly improved.

1. INTRODUCTION

Structure from Motion (SfM) aims to reconstruct a 3D scene from a set of images and estimate the camera poses. It has been implemented in well-known software packages, such as Bundler (Snavely et al., 2007), COLMAP (Schonberger and Frahm, 2016), and applied in 3D modeling of internet images (Geppert et al., 2020) and Unmanned Aerial Vehicle (UAV) images (Jiang et al., 2020) (Li et al., 2023). The general procedure of SfM includes feature extraction, feature matching, view graph construction and 3D reconstruction (Jiang et al., 2022b).

A view graph is an undirected weighted graph, where vertexes represent images and edges represent connections among images, and it is the input of 3D reconstruction (Jiang and Jiang, 2017) (Liu et al., 2022). The construction of the view graph depends on the results of feature matching. But the computation of feature matching is expensive, especially for large-scale UAV datasets, which have a large number of images with high spatial resolution. The exhaustive matching method cannot satisfy the system's requirement for efficiency. To tackle this problem, image retrieval is used to compute a matching candidate subset to accelerate feature matching (Jiang and Jiang, 2020). The current SfM systems, like COLMAP, use vocabulary tree (Nister and Stewenius, 2006) to acquire similar image pairs. As datasets grow larger, this method also gradually fails to meet the requirement due to high time consumption in the indexing of large-size of local features. Thus, more discriminative small-size global features and efficient approximate nearest-neighbor (ANN) searching methods provide new solutions.

In the field of image retrieval, Bag of Word (BoW) (Sivic and Zisserman, 2003), Fisher Vector (FV) (Perronnin et al., 2010), and Vector of Locally Aggregated Descriptors (VLAD) (Jegou et al., 2012) are the most typical methods for image vector representation. In recent years, learning-based image retrieval has also developed rapidly. The 3D model obtained from SfM was used by Radenović et al. (2016) to guide the selection of the training data for fine-tuning the network model, which aggregates features with a max pooling layer and is suitable for the task of image retrieval. Inspired by VLAD, a differentiable NetVLAD layer was proposed and embedded into Convolutional Neural Network (CNN) models for image retrieval (Arandjelovic et al., 2018). In order to enhance the performance of image retrieval, Radenović et al. (2019) proposed a novel trainable generalized mean (GeM) pooling layer that generalizes max pooling and average pooling.

The accuracy and efficiency of image retrieval are crucial to view graph construction. In addition, it's important to avoid the loss of too many correct matching pairs, which would damage the completeness of the view graph and reconstruction. Efficient and accurate image retrieval methods support constructing a stable view graph to complete the reconstruction.

In this paper, we describe view graph construction and image retrieval methods in Section 2. Section 3 gives details of the evaluation metrics and datasets in tests. Finally, we present and discuss the results of our experiments in Section 4.

2. STATE-OF-THE-ART METHODS FOR VIEW GRAPH CONSTRUCTION

2.1 View Graph Construction

A view graph is required to guide the process of reconstruction in SfM. View graph is represented as an undirected weighted graph $G = (V, E)$, where a vertex in the vertex set V represents an image and an edge in the edge set E represents a connection for an image pair (Jiang et al., 2022c) with an assigned weight. In addition, the relative geometries of the image pairs are preserved for the edges. After obtaining a matching candidate subset by image retrieval, filtering with feature matching and

https://doi.org/10.5194/isprs-archives-XLVIII-1-W2-2023-1059-2023 | © Author(s) 2023. CC BY 4.0 License.

geometric verification is performed on the subset to derive the final matching pairs. Assume that the image set $A = \{a_i\}$ has N images and the number of matching pairs in set $B = \{b_{ij}\}$ is M. The construction of the view graph G is as follows: firstly, adding the vertex v_i to set V for a_i, where $i = 1, \dots, N$. Then, adding the edge e_{ij} connecting v_i and v_j to set E for b_{ij} connecting a_i and a_j. Finally, weights are assigned to the edges to differentiate the importance of the edges. Assume that the set of weights is $W = \{w_{ij}\}$. The weight w_{ij} depends on the number of matching features and their distribution on the image, as in formula (1), where R_{ew} is the weight ratio between w_{inlier} and $w_{overlap}$, w_{inlier} and $w_{overlap}$ are determined by the number of matching features and the distribution of matching features, respectively.

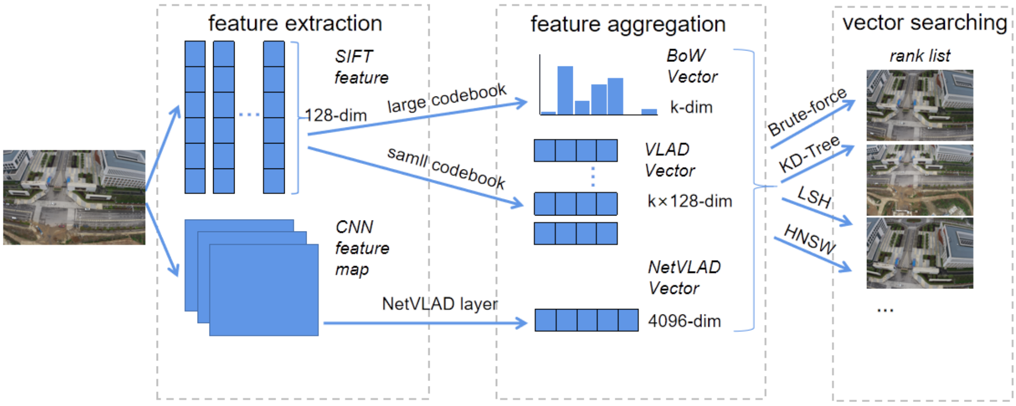

Figure 1. The pipeline of the retrieval methods. SIFT features or CNN feature maps of the input image are extracted in the feature extraction stage. SIFT features are transformed into BoW or VLAD vector based on different aggregation methods and codebooks, and CNN feature maps are transformed into NetVLAD vector through a NetVLAD layer. Four vector searching methods can be used to obtain the rank list.

$$w_{ij} = R_{ew} \times w_{inlier} + (1 - R_{ew}) \times w_{overlap} \qquad (1)$$

To address the low efficiency of incremental SfM, the parallel technique has been used, in which the created view graph can be utilized to guide the scene partition and merging. Generally, the Normalized Cut algorithm (Shi and Malik, 2000) removes the edges with lower weights in the view graph and divides the entire scene into several clusters. These clusters are reconstructed in parallel and the cluster reconstructions are merged to obtain the final reconstruction model.

2.2 Image Vector Representation

The general method of image retrieval is first to extract vector representations of images in a database. Then, the query image is also converted into a vector, and the top-k nearest vectors are found by Nearest Neighbour (NN) searching. Finally, the images corresponding to the vectors are the retrieval results. Image vector representation includes traditional methods and learning-based methods. Traditional methods, such as BoW (Sivic and Zisserman, 2003) and VLAD (Jegou et al., 2012), generate global features from hand-crafted local features, like Scale Invariant Feature Transform (SIFT) (Lowe, 2004). Some researchers have attempted to apply deep learning to image retrieval due to its impressive performance on tasks like image classification. The common approach is to design a feature aggregation layer and add it behind the last convolution layer in existing CNN architectures. Then fine-tuning the new network so that it can be adapted to the task of image retrieval, NetVLAD (Arandjelovic et al., 2018) is one of the representatives. The retrieval pipeline shown in Figure 1, includes three feature aggregation methods and four vector searching methods.

2.2.1 BoW: By treating images as documents and local features as words, BoW, a popular algorithm in the field of information retrieval, can be used to retrieve images. In image retrieval, the visual codebook is usually generated through K-means clustering. With the codebook, images can be converted into histogram vectors. A dimension of the vector corresponds to a cluster center. The value of the dimension is equal to the number of local features that are close to the corresponding cluster center. BoW-based image retrieval has been the most classic method for finding matching candidate subsets (Jiang et al., 2022a). For large-scale UAV datasets, a large-size codebook is required to ensure the differentiation of BoW vectors, which results in a long vector production process and high vector dimensionality. Additionally, a significant amount of information on local features is lost as a result of the counting mechanism.

2.2.2 VLAD: The same as BoW, VLAD also requires a trained codebook and local features from feature extraction. Assume that N is the number of local features, D is the local feature dimension, and K is the number of codebook centers. This is how the VLAD descriptor is calculated: for each local feature, finding its nearest cluster center. Then we calculate the residuals between the cluster centers and the local features and sum the residuals for all cluster centers. This transforms the N × D initial image feature into a K × D VLAD descriptor. Formula (2) shows the specific calculation of VLAD, where $v_{k.j}$ indicates the jth dimension of the kth row of the VLAD descriptor, $x_{i,j}$ indicates the jth dimension of the ith local feature, $c_{k,j}$ indicates the jth dimension of the kth cluster center, and a is an assignment function. When the nearest cluster center of x_i is c_k, $a_{i,k} = 1$, otherwise $a_{i,k} = 0$.

$$v_{k.j} = \sum_{i=1}^{n} a_{i,k}(x_{i,j} - c_{k,j}) \qquad (2)$$

In general, the size of the codebook used for VLAD is much smaller than that of BoW because it retains the dimension of the local feature. Besides, the size of cluster centers K is also much smaller than the number of extracted features N from UAV images. Thus, high efficiency can be achieved for image indexing by using the VLAD vector.

2.2.3 NetVLAD: A differentiable NetVLAD layer is designed since the indifferentiable assignment function a in the initial VLAD is not available for the training of CNN, and the new assignment function is shown in formula (3), where x_i denotes a local feature, c_k denotes a cluster center, and α is a positive parameter. Instead of assigning a local feature to only one clustering center, the assignment values of a local feature to different cluster centers are determined by the distances between them.

$$\bar{a}_k(x_i) = \frac{e^{-\alpha\|x_i - c_k\|^2}}{\sum_{k'} e^{-\alpha\left\|x_i - c_{k'}\right\|^2}} \tag{3}$$

Formula (4) is obtained by expanding formula (3) and dividing the numerator and denominator with $e^{-\alpha\|x_i\|^2}$, where $w_k = 2\alpha c_k$, $b_k = -\alpha\|c_k\|^2$. It is a softmax function.

$$\bar{a}_k(x_i) = \frac{e^{w_k^T x_i + b_k}}{\sum_{k'} e^{w_{k'}^T x_i + b_{k'}}} \tag{4}$$

Therefore, the NetVLAD descriptor is calculated as in formula (5), where w_k and b_k are trained to obtain the assignment values and the parameter c_k is trained as the cluster center. Since the cluster centers are acquired through supervised learning, they are preferable to those acquired through clustering in traditional methods, making the NetVLAD descriptor more distinctive.

$$v(j,k) = \sum_{i=1}^{N} \frac{e^{w_k^T x_i + b_k}}{\sum_{k'} e^{w_{k'}^T x_i + b_{k'}}} (x_i(j) - c_k(j)) \tag{5}$$

The strong representation capability of deep learning enables the NetVLAD descriptor to perform well in image retrieval. Through embedding the NetVLAD layer into MobileNet and compressing the network model by knowledge distillation, NetVLAD descriptors have been used for image retrieval-based visual localization on mobile platforms (Sarlin et al., 2019).

2.3 Nearest Neighbor Searching

After transforming images into vector representations, we find the top-k nearest vectors of the query vector by nearest neighbor (NN) searching. Brute-force searching is the simplest and highest time complexity method, and KD-Tree (Bentley, 1975) is an NN searching algorithm like it to find the accurate nearest neighbors. Many approximate nearest neighbor (ANN) searching algorithms have been proposed because of the high time cost of NN searching, they all improve the searching efficiency at the expense of accuracy. For example, graph-based Hierarchical Navigable Small World (HNSW) (Malkov and Yashunin, 2020), and hashing-based Locality-Sensitive Hashing (LSH) (Indyk and Motwani, 1998).

2.3.1 Brute-force: Brute-force searching traverses all vectors in the database, calculates the distances between the query vector and them, and ultimately determines the top-k nearest neighbors of the query vector by sorting vectors in the database according to the distances. This approach is not suitable for the case of high vector dimension and large amounts of data, and its result is generally utilized as a benchmark to evaluate other methods.

2.3.2 KD-Tree: KD-Tree is a data structure for partitioning K-dimensional data space and is widely used for NN searching. The K-dimensional space is divided into numerous subspaces through iterative partitioning, which is based on the largest-variance dimension. The nearest neighbors of a query vector are discovered by dichotomous searching with a traceback mechanism after constructing a KD-Tree. As a result of the low dimension of the utilized feature descriptors, like the SIFT descriptor with 128-dimension, KD-Tree has been widely used for image retrieval algorithms (Hu and Nooshabadi, 2019; Huang et al., 2010) and software packages, like COLMAP (Schonberger and Frahm, 2016) and AliceVision (Griwodz et al., 2021). As the feature dimension increases, the efficacy of KD-Tree declines and even falls below that of brute-force searching.

2.3.3 LSH: It is time-consuming to perform brute-force searching and KD-Tree searching on massive high-dimensional datasets, this issue is partially resolved by LSH. The main idea of LSH is that the probability of two vectors in the original vector space that are close to one another being hashed into the same cell is still high after the same mapping; in contrast, the probability is small when the distance between two vectors is large. After indexing the vectors with a locality-sensitive hashing function, the function is used on the query vector to find the corresponding cell at the time of searching, then the query vector performs a similarity measure with the vectors in this cell to find its approximate nearest neighbors. LSH has been employed for large-scale image retrieval, including online community and remote sensing photos, because of its high efficiency (Li et al., 2021).

2.3.4 HNSW: HNSW and Navigable Small World (NSW) (Malkov et al., 2014) are graph-based algorithms in the field of ANN searching. The Small World is a kind of graph between the regular graph and the random graph, in which the connections of the same class of nodes are locally regular, and the connections of different classes of nodes are globally random. The construction of NSW is similar to its query operation and based on a modified naive Proximity Graph algorithm. In contrast to the Delaunay graph, it reduces the time complexity of construction and adds "Highways" with randomness. There are long edges and short edges in NSW. The approximate Delaunay graph for greedy searching is composed of short edges. The long edges, known as "Highways", are used for the logarithmic scaling of the greedy searching and result in the graph with the property of Navigable Small World. A multi-layer framework is built in HNSW, which is an enhanced variant of NSW, to improve the efficiency of ANN searching. HNSW was used by Liu et al. (2022) to substitute KD-Tree, and it enhances the efficiency of image retrieval.

3. EVALUATION METRICS AND DATASETS

3.1 Evaluation Metrics

The accuracy of image retrieval is important for view graph construction. It is calculated by the retrieval results and the ground truth derived from the reconstruction results based on sufficient matching pairs. The definition of the accuracy is shown in formula (6), where $N(*)$ indicates the number of image pairs in the set $*$, GT is the set of ground truth and IR is the set of image retrieval results.

$$Accuracy = \frac{N(IR \cap GT)}{N(GT)} \tag{6}$$

It is also significant to evaluate the efficiency as we aim to shorten the time consumption of image retrieval to speed up feature matching. The efficiency is given in formula (7), where T_{vec} denotes the elapsed time of obtaining image vector representations and T_{nns} denotes the elapsed time of NN searching.

$$Efficiency = T_{\text{vec}} + T_{\text{nns}} \tag{7}$$

Some statistical metrics of the reconstruction model also reflect the quality of the view graph constructed with different image retrieval methods. The number of registered images and points indicates the completeness of the model, while the mean reprojection error indicates the precision of the model. As the final task is the reconstruction of UAV images, it is necessary to consider these metrics.

3.2 Datasets

The performance of the image retrieval methods is evaluated by three different scale UAV datasets, Campus, SZU, and 2W, which were captured at China University of Geosciences, Shenzhen University, and a township area, respectively. Details of the datasets are presented in Table 1, and Figure 2 shows a sample image of each dataset. Among Table 1, the most influential on the retrieval performance is the number of images and the image size.

Item	Campus	SZU	2W
UAV type	multi-rotor	multi-rotor	multi-rotor
Flight height(m)	80	-	87.1
Camera mode	DJI FC6310R	DJI Zenmuse P1	SONY ILCE 7R
Number of cameras	1	1	5
Focal length(mm)	24	35	35
Camera angle(°)	0	-	Nadir: 0; oblique: 45/-45
Number of images	3,743	4,030	21,654
Image size(pixel)	5472×3648	8192×5460	6000×4000
GSD(cm)	2.6	1.2	1.21

Table 1. Details of the three datasets.

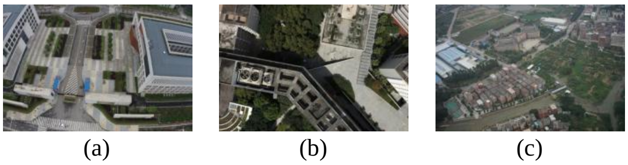

Figure 2. Sample images of the three datasets: (a) Campus; (b) SZU; (c) 2W.

4. EXPERIMENTAL RESULTS AND DISCUSSION

In this section, the evaluation experiments of image retrieval methods are performed on the three datasets described in Section 3. Then the optimal method is applied for view graph construction and reconstruction of the datasets.

4.1 Evaluation of Image Retrieval

There are three image vector representation methods BoW, VLAD, NetVLAD, and four searching methods Brute-force, KD-Tree, LSH, and HNSW in our experiments. The efficiency and accuracy of these approaches are shown in Table 2 and Table 3, respectively. In the tables, the first column lists the vector representation approaches, the second column lists the searching approaches, and the final three columns show the results on the three datasets.

The local features used in experiments BoW and VLAD are 128-dimension SIFT descriptors extracted from feature extraction in SfM. The network model used in experiments NetVLAD is VGG16 and is fine-tuned with dataset Pitts30k. The number of clustering centers in experiments BoW, VLAD64, VLAD256, and NetVLAD is 256 × 256, 64, 256, and 64, respectively. In experiments HNSW10, there are 10 connections for each node, which decreases accuracy on dataset 2W. Therefore, we changed the number of connections to 64 and performed the HNSW64 experiments. The following is a discussion and conclusion of the results.

In the results of experiments BoW and VLAD64, the high accuracy of BoW is mainly because the number of cluster centers is much larger than that of VLAD64, which also leads to lower efficiency. Despite the highest efficiency of LSH, it makes the accuracy drop a lot. While ensuring accuracy, HNSW10 increases the searching efficiency and is only slightly less efficient than LSH. VLAD64 with HNSW10 is approximately 120 times faster than BoW, and the accuracy is within 0.07 lower than BoW. We raised the number of VLAD cluster centers to improve accuracy, the accuracy was significantly enhanced and even surpassed BoW, as shown in VLAD256 in Table 2. As the number of cluster centers grows, efficiency inevitably decreases, but VLAD256 with HNSW10 or HNSW64 is still roughly 100 times faster than BoW and also faster than NetVLAD.

Vector	Searching	Campus	SZU	2W
BoW	KD-Tree	84.29	**78.54**	80.08
VLAD 64	Brute-force	80.10	71.78	76.54
	KD-Tree	80.10	71.78	76.54
	LSH	76.21	69.20	73.54
	HNSW10	81.71	72.85	75.56
	HNSW64	-	-	77.05
VLAD 256	Brute-force	86.30	77.44	82.11
	KD-Tree	86.30	77.44	82.11
	LSH	80.45	71.88	76.89
	HNSW10	**87.30**	77.99	78.68
	HNSW64	-	-	**82.18**
Net VLAD	Brute-force	78.93	75.99	76.99
	KD-Tree	78.93	75.99	76.99
	LSH	76.87	74.58	75.42
	HNSW10	79.52	75.99	75.33
	HNSW64	-	-	77.05

Table 2. Accuracy comparison of combinations of vector representation and nearest neighbor searching methods on the three datasets. The bold indicates the highest accuracy on each dataset.

Then comes the learning-based NetVLAD, which is less than the accuracy of BoW but higher than the VLAD64 with the same number of cluster centers on dataset SZU and 2W. It is about 8 times more efficient than BoW. The main reason why the efficiency of NetVLAD is not as good as VLAD is that while VLAD utilizes local features extracted in the previous step, NetVLAD must extract deep features as local features individually. Furthermore, the resolution of an image is also a significant factor affecting the efficiency of NetVLAD. The deep feature extraction of a larger resolution image takes more time. For instance, dataset SZU has 287 more images than dataset Campus but takes twice as long as dataset Campus. Although NetVLAD does not perform as well as VLAD256, it is significantly more efficient than BoW and outperforms VLAD64 with the same number of cluster centers in terms of accuracy.

Vector	Searching	Campus	SZU	2W
BoW	KD-Tree	5476.24	5838.51	74141.48
VLAD 64	Brute-force	154.35	177.74	4062.51
	KD-Tree	124.27	143.31	3548.91
	LSH	**43.14**	**49.16**	**257.28**
	HNSW10	44.02	50.14	262.99
	HNSW64	-	-	289.24
VLAD 256	Brute-force	556.89	663.76	9523.80
	KD-Tree	390.93	402.69	13886.88
	LSH	55.47	69.52	451.52
	HNSW10	59.97	74.62	478.23
	HNSW64	-	-	526.91
Net VLAD	Brute-force	426.15	766.40	5746.23
	KD-Tree	385.22	717.62	4348.38
	LSH	342.07	668.07	2833.54
	HNSW10	342.41	668.45	2835.14
	HNSW64	-	-	2839.65

Table 3. Efficiency comparison of combinations of vector representation and nearest neighbor searching methods on the three datasets. The bold indicates the most efficient method (in seconds) on each dataset.

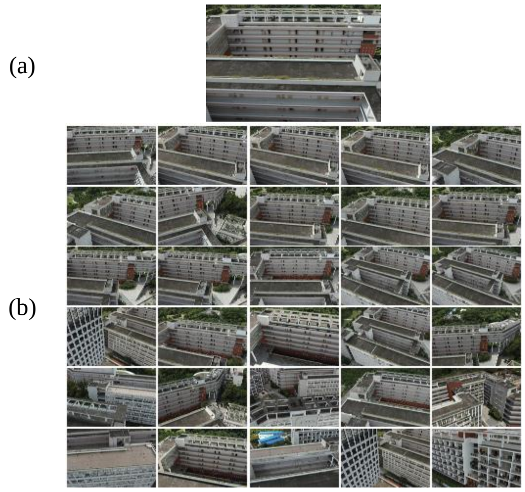

Figure 3. A retrieval instance of VLAD256 with HNSW on dataset SZU. (a) query image; (b) retrieved images.

As a result, the optimal vector representation approach in our experiments is VLAD256, which has the highest or second-highest accuracy and is only less efficient than VLAD64. The best searching approach is HNSW, which is comparable to the fastest LSH in terms of efficiency and without decreasing the retrieval accuracy as long as the parameter is properly adjusted. Figure 3 shows a retrieval instance of the optimal methods on dataset SZU. Afterward, we used VLAD256 with HNSW to obtain matching candidate subsets for constructing view graphs that guided the reconstructions of the three datasets.

4.2 View Graph Construction and SfM Reconstruction

The range of feature matching can be restricted via the matching candidate subset from image retrieval. Then a view graph was constructed for each dataset based on the feature matching results. Figure 4 shows the view graph of dataset SZU, where the red dots indicate the nodes and the gray lines indicate the connections. All the images are added to the view graph, and 67787 matching pairs are retained.

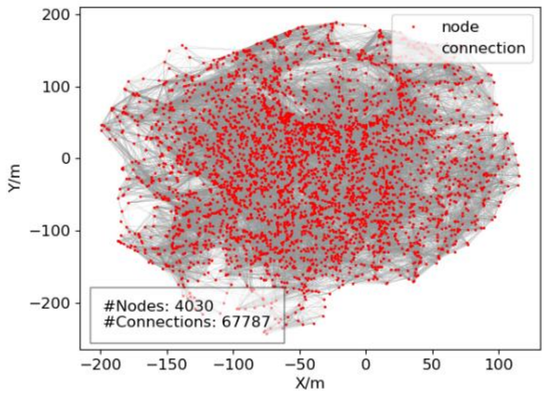

Figure 4. The view graph of dataset SZU.

Before reconstruction, the view graph was partitioned into small clusters with strong connections intra-clusters and weak connections inter-clusters by the normalized cut algorithm. The partitioned view graph is composed of 9 clusters, and the maximum number of images in each cluster is 500, as in Figure 5, where one color represents one cluster.

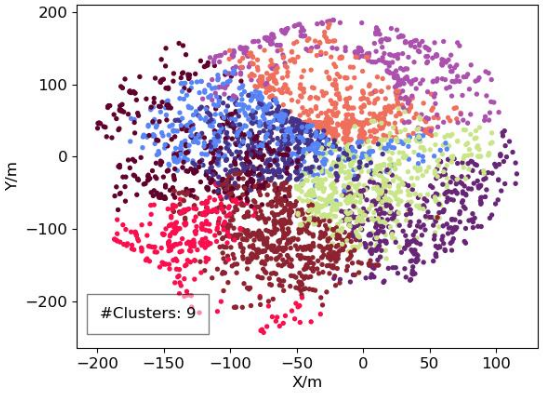

Figure 5. The partitioned view graph of dataset SZU.

We performed incremental reconstruction for the clusters in parallel. Then, the cluster reconstructions were merged to obtain the complete reconstruction model. Figure 6 presents the reconstruction model of dataset SZU and the statistics of the reconstruction model of the three datasets are given in Table 4. The mean reprojection error, which manifests the precision of the model, is 0.702 for Campus, 0.802 for SZU, and 0.954 for

2W. The number of registered images and the number of reconstructed 3D points reflect the completeness of the model, and almost all the images are registered for each dataset. It follows that the optimal methods can construct a robust view graph to accomplish reconstruction.

	Campus	**SZU**	**2W**
Registered images	3737	4029	21642
Points	978395	1504264	8943192
Mean reprojection error(pixel)	0.702	0.802	0.954

Table 4. Reconstruction model statistics of the three datasets.

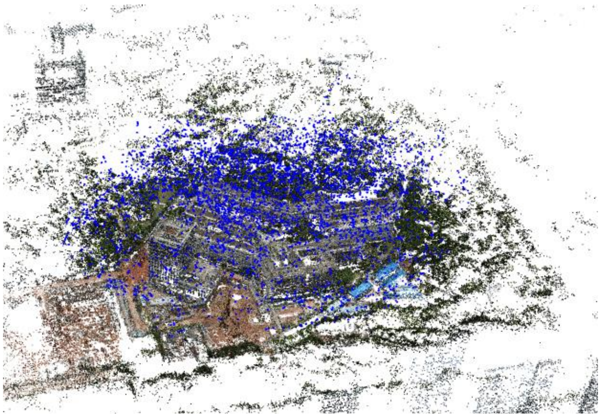

Figure 6. The reconstruction model of dataset SZU.

5. CONCLUSIONS AND FUTURE STUDIES

In this paper, we compared different image vector representation methods, both the traditional VLAD descriptor and the learning-based NetVLAD descriptor outperform the vocabulary tree approach in existing SfM systems in terms of efficiency. For accuracy, NetVLAD and VLAD descriptors with 64 cluster centers are lower than the vocabulary tree approach, but the VLAD descriptor with 256 cluster centers outperforms it. We also compared different NN searching and ANN searching methods, and graph-based HNSW significantly outperforms the other methods. Utilizing the optimal scheme VLAD256 with HNSW to select a matching candidate subset for view graph construction, large-scale UAV images can be reconstructed efficiently and accurately. The experiment results demonstrate that the optimal scheme can accelerate the matching candidate subset searching approximately 100 times and also improve the efficiency of SfM reconstruction.

The local features used in this study are hand-crafted descriptors, which can be replaced with powerful learning-based descriptors in future studies. Besides, as more capable deep learning models are proposed, embedding NetVLAD into them and training them for the task of 3D reconstruction may improve accuracy and efficiency.

ACKNOWLEDGEMENTS

This research was funded by the National Natural Science Foundation of China (Grant No. 42001413), the Open Research Fund from the Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ) (Grant No. GML-KF-22-08), and the Open Research Project of The Hubei Key Laboratory of Intelligent Geo-Information Processing (Grant No. KLIGIP-2021B11).

REFERENCES

Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J., 2018. NetVLAD: CNN Architecture for Weakly Supervised Place Recognition. IEEE Trans Pattern Anal Mach Intell 40, 1437-1451.

Bentley, J.L., 1975. Multidimensional binary search trees used for associative searching. Communications of the ACM 18, 509-517.

Geppert, M., Larsson, V., Speciale, P., Schönberger, J.L., Pollefeys, M., 2020. Privacy preserving structure-from-motion, Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16. Springer, pp. 333-350.

Griwodz, C., Gasparini, S., Calvet, L., Gurdjos, P., Castan, F., Maujean, B., De Lillo, G., Lanthony, Y., 2021. AliceVision Meshroom, Proceedings of the 12th ACM Multimedia Systems Conference, pp. 241-247.

Hu, L., Nooshabadi, S., 2019. High-dimensional image descriptor matching using highly parallel KD-tree construction and approximate nearest neighbor search. Journal of Parallel and Distributed Computing 132, 127-140.

Huang, K.-Y., Tsai, Y.-M., Tsai, C.-C., Chen, L.-G., 2010. Video stabilization for vehicular applications using SURF-like descriptor and KD-tree, 2010 IEEE International Conference on Image Processing. IEEE, pp. 3517-3520.

Indyk, P., Motwani, R., 1998. Approximate nearest neighbors: towards removing the curse of dimensionality, Proceedings of the thirtieth annual ACM symposium on Theory of computing, pp. 604-613.

Jegou, H., Perronnin, F., Douze, M., Sanchez, J., Perez, P., Schmid, C., 2012. Aggregating local image descriptors into compact codes. IEEE Trans Pattern Anal Mach Intell 34, 1704-1716.

Jiang, S., Jiang, C., Jiang, W., 2020. Efficient structure from motion for large-scale UAV images: A review and a comparison of SfM tools. ISPRS Journal of Photogrammetry and Remote Sensing 167, 230-251.

Jiang, S., Jiang, W., 2017. Efficient structure from motion for oblique UAV images based on maximal spanning tree expansion. ISPRS Journal of Photogrammetry and Remote Sensing 132, 140-161.

Jiang, S., Jiang, W., 2020. Efficient match pair selection for oblique UAV images based on adaptive vocabulary tree. ISPRS Journal of Photogrammetry and Remote Sensing 161, 61-75.

Jiang, S., Jiang, W., Guo, B., 2022a. Leveraging vocabulary tree for simultaneous match pair selection and guided feature matching of UAV images. ISPRS Journal of Photogrammetry and Remote Sensing 187, 273-293.

Jiang, S., Jiang, W., Wang, L., 2022b. Unmanned Aerial Vehicle-Based Photogrammetric 3D Mapping: A survey of techniques, applications, and challenges. IEEE Geoscience and Remote Sensing Magazine 10, 135-171.

https://doi.org/10.5194/isprs-archives-XLVIII-1-W2-2023-1059-2023 | © Author(s) 2023. CC BY 4.0 License.

Jiang, S., Li, Q., Jiang, W., Chen, W., 2022c. Parallel Structure From Motion for UAV Images via Weighted Connected Dominating Set. IEEE Transactions on Geoscience and Remote Sensing 60, 1-13.

Li, Q., Huang, H., Yu, W., Jiang, S., 2023. Optimized Views Photogrammetry: Precision Analysis and a Large-Scale Case Study in Qingdao. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 16, 1144-1159.

Li, X., Yang, J., Ma, J., 2021. Recent developments of content-based image retrieval (CBIR). Neurocomputing 452, 675-689.

Liu, S., Jiang, S., Liu, Y., Xue, W., Guo, B., 2022. Efficient SfM for Large-Scale UAV Images Based on Graph-Indexed BoW and Parallel-Constructed BA Optimization. Remote Sensing 14.

Lowe, D.G., 2004. Distinctive image features from scale-invariant keypoints. International journal of computer vision 60, 91-110.

Malkov, Y., Ponomarenko, A., Logvinov, A., Krylov, V., 2014. Approximate nearest neighbor algorithm based on navigable small world graphs. Information Systems 45, 61-68.

Malkov, Y.A., Yashunin, D.A., 2020. Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs. IEEE Trans Pattern Anal Mach Intell 42, 824-836.

Nister, D., Stewenius, H., 2006. Scalable recognition with a vocabulary tree, 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). Ieee, pp. 2161-2168.

Perronnin, F., Liu, Y., Sánchez, J., Poirier, H., 2010. Large-scale image retrieval with compressed fisher vectors, 2010 IEEE computer society conference on computer vision and pattern recognition. IEEE, pp. 3384-3391.

Radenović, F., Tolias, G., Chum, O., 2016. CNN image retrieval learns from BoW: Unsupervised fine-tuning with hard examples, Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer, pp. 3-20.

Radenović, F., Tolias, G., Chum, O., 2019. Fine-Tuning CNN Image Retrieval with No Human Annotation. IEEE Trans Pattern Anal Mach Intell 41, 1655-1668.

Sarlin, P.-E., Cadena, C., Siegwart, R., Dymczyk, M., 2019. From Coarse to Fine: Robust Hierarchical Localization at Large Scale, 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 12708-12717.

Schonberger, J.L., Frahm, J.-M., 2016. Structure-from-Motion Revisited, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4104-4113.

Shi, J., Malik, J., 2000. Normalized cuts and image segmentation. IEEE Transactions on pattern analysis and machine intelligence 22, 888-905.

Sivic, J., Zisserman, A., 2003. Video Google: A text retrieval approach to object matching in videos, Computer Vision, IEEE International Conference on. IEEE Computer Society, pp. 1470-1470.

Snavely, N., Seitz, S.M., Szeliski, R., 2007. Modeling the World from Internet Photo Collections. International Journal of Computer Vision 80, 189-210.

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-1-W2-2023-1059-2023 | © Author(s) 2023. CC BY 4.0 License.