A Sequence-Based Visual Place Recognition Technique with Segmented Database and Compact Sequence List

Xin-Zhuo Li[1], Cong Li[2], and Hung-Chyun Chou[3]

***Abstract*— Sequence-based visual place recognition algorithms have been proven to be able to handle environmental changes caused by illumination, weather, and time of the day with handcrafted descriptors. However, an exhaustive search for all images in a query sequence is computationally expensive. In this paper, we propose a technique that can significantly reduce the size of searching space for sequence matches while remaining state-of-the-art accuracy. Firstly, we managed to achieve a better selection of reference candidates of images in query sequence by segmenting the database according to similarity. Then, a much more informative and compact query sequence is designed by removing all the unnecessary images in the original query sequence. State-of-the-art performance is reported on a public dataset with challenging environmental changes. Our algorithm shows comparable accuracy with other current best results and exceeds all the other methods in the dataset with illumination variation. In addition, the decrease in execution time and higher success rate for selecting candidates of query images for sequence match is also provided.**

***Index Terms*— Challenging environmental changes, informative and compact query, place recognition**

I. Introduction

Visual place recognition (VPR) as a sub-domain of simultaneous localization and mapping (SLAM) has been widely discussed in recent years because of its contribution to the loop closure step in SLAM system. Recognizing a previously visited place correctly is indispensable for subsequent optimization. In general, VPR can be considered as an image retrieval problem of finding the most similar matches of the query single-frame or sequence of frames in the reference database built by the previous traverse. However, viewpoint variation, varied illumination, and weather conditions could lead to the different appearances of the same place which increase the difficulty of the recognition task significantly. Currently, CNN-based VPR techniques such as [1] [2] [3] have succeeded on the most challenging VPR datasets, as evaluated in [4] and [5]. However, to train a CNN for VPR tasks, large-scale datasets from different environments under various angles, seasons and illumination conditions are needed. Moreover, the CNN's encoding time and runtime memory are much higher than those required for the handcrafted descriptors. Although the CNN-based VPR techniques have outperformed handcrafted descriptor-based techniques currently, their intense computational requirement is still a serious problem.

Instead of using CNN-based VPR algorithms, a sequence-based method has also demonstrated impressive performance in recognizing places that underwent severe appearance changes with handcrafted descriptors. In SeqSLAM [6], much better performance has been shown compared to previous single-frame SLAM algorithms like FAB-MAP 2.0 [7]. However, exhaustive sequence search is computationally costly in a large database. The sequence match procedure basically contained two steps. The first step is evaluating all the reference images in database to generate a distance vector for every query frame in sequence and combining all the distance vectors to get a holistic distance matrix. And the second step is finding the best matching sequence in that distance matrix. To reduce the computational time in first step, retrieval techniques based on data structures like trees [8][9], graphs [10][11][12] and hashing [13][14][15] has been used in previous research. As for the second step, we can either select a limited number of reference candidates for each query frame in sequence or set dynamic length of the sequence to reduce the size of the distance matrix. Our method integrates techniques of both steps together and use the information generated from first step to achieve a better performance in second step with no extra computation.

In this paper, we propose a novel VPR solution by utilizing the similarity of images to segment database and simplify query sequence. Firstly, a hierarchical retrieval in segmented database for each frame in query sequence is implemented to reduce single-frame searching time. As for sequence match, candidates of each query image will be selected to avoid exhaustive search in whole database. The segmentation of database also helps to exclude clustering incorrect reference images in candidates' selection. Unnecessary interval images which can provide little supplementary information in a sequence are also ignored to achieve a more compact query list. After all the manipulation, the searching space for sequence match can be reduced significantly to achieve computational efficiency. Additionally, most of previous sequence-based algorithms assume a constant velocity between consecutive samples of reference and query data, which makes them unreliable in practice when velocity changed, or robot kidnapping occurred. In our algorithm, such assumption is not needed. Instead, we use the similarity between images to ensure the same distance of adjacent query frames in sequence and their corresponding reference images. In other words, the constraint we use to find sequence in database is from similarity calculation instead of invariant velocity assumption.

In general, the main contributions in this paper are listed below:

- Use similarity between images to segment database to achieve a hierarchical searching method.

*This work was supported by National Key R&D Program of China No. U2013202, the Guangdong Basic and Applied Basic Research Foundation under Grant No. 2022A1515011139, and the Shenzhen Science and Technology Program Grant No. JSGG20210802152801004.

*Corresponding author: Hung-Chyun Chou, Email: zhouhongjun@cuhk.edu.cn

[1]Xin-Zhuo Li is with Beijing Jiaotong University, Beijing, 100044, China.

[2]Cong Li is with Department of Mechanical Engineering and Materials Science, Duke University, Durham, NC 27708, USA.

[3]Hung-Chyun Chou is with Special Robot Center, Shenzhen Institute of Artificial Intelligence and Robotics for Society, 14-15F, Tower G2, Xinghe World, Rd Yabao, Longgang District, Shenzhen, Guangdong, 518129, China.

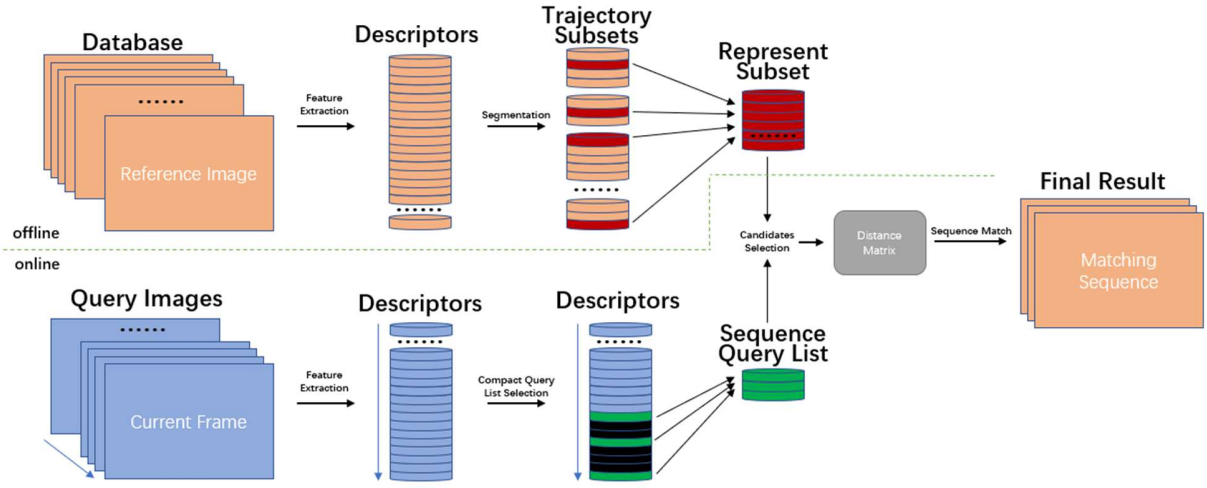

Fig. 1. The general procedure of our sequence based VPR technique

- Better performance of candidate's selection for query frames in sequence with the help of database segmentation.
- Significantly simplify sequence match by removing all the unnecessary interval images in query sequence to achieve a much more compact query list.
- Utilize the same similarity threshold for both database and query images to avoid using searching method based on invariant velocity assumption.

The rest of the paper is organized as follows. In Section II, we summarize relevant prior research in VPR. Section III describes our algorithm, both the single-frame retrieval and sequence match method will be illustrated. Our experimental design and results are presented in Section IV, focusing on the accuracy of our algorithm and the reduction of execution time.

II. Related Work

After SeqSLAM [6] was proposed, many following research have been done to solve the high computational cost problem. For example, Fast-SeqSLAM [16] used an approximate nearest neighbor algorithm [17] to search for u nearest neighbor images for a current view where u is constant and much smaller than the number of images in the database. Then, a sparse matrix is created for subsequent sequence match. For each frame in the query list, only a few candidates remain in this matrix. As in DOSeqslam [18], the incoming image stream is segmented to using loop closure detection techniques to decrease the searching domain in the database. In their method, whether two images share a common local descriptor is used to evaluate similarity between images, which leads to a dynamic sequence length. While in ConvSequential-SLAM [19], the author proposed a concept of information gain to evaluate whether the sequence length is enough for matching. They utilized Histogramof-Oriented-Gradients (HOG) descriptors and entropy map representing the salient regions in each query image to compute the information gain. New frames will be added to the sequence successively until the sequence reaches required information gain. This also helped them to achieve a dynamic sequence length. In SeqNet [20], short sequential descriptors are learned to generate high-performance initial match candidates. The descriptor is not related to a single frame but a sequence of frames which also helps to reduce the length of the query sequence and improve the matching accuracy. In DeepSeqSLAM [21], a trainable CNN and RNN architecture is proposed to apply sequence matching on deep learning models to solve VPR problems. Even though its accuracy outperforms all the other algorithms, it requires too many computational resources in practice.

Even though all these algorithms managed to be more computationally efficient compared to SeqSLAM by reducing the size of searching space, they are still based on the invariant velocity assumption. Moreover, similar adjacent images in query sequence which can provide little supplementary information for retrieval were being exhaustively matched in their methods. Although dynamic sequence length has been used in previous research because the performance of sequence match is strongly related to sequence length, the fact that similar query images in a sequence are abundant has not been realized yet.

III. Methodology

This section presents the methodology proposed in our work, including modified information gain computation method, segmentation of database, hierarchical retrieval strategy based on segmentation, simplification of query sequence and how to implement sequence match with previously generated features. The general procedure is shown in Fig. 1. For the offline processes, the descriptors of reference images in database will firstly be calculated. Secondly, the database will be segmented by putting similar adjacent images in one subset. Each subset represents the trajectory of the traverse. Then the most informative image in each trajectory subset which can represent all images in this set will be gathered into the represent subset whose size is much smaller than the database. For the online processes, the descriptors of query images will first be extracted as well. After that, the query sequence for the current frame is created by adding images before it with enough different features iteratively to obtain a list of size M. Only the green images in the graph will be added to the sequence, the black images are ignored because they are like the current frame which can provide little additional information for the holistic sequence. Then, each query image will search their best match in represent subset to find best N images. Further search in N trajectory subsets that are represented by these best N images will be carried out. Each trajectory subset will generate a local best

match. These local best matches are gathered as candidates for query images. A distance matrix D of size $M \times N$ that contains the similarity score between reference and query images is created subsequently. After sequence match in space D, we can obtain the final matching sequence. Detailed information is provided by the following subsections.

A. Similarity and Information Gain

To obtain the similarity between images, we applied a computational-efficient and training-free approach called CoHoG [22]. Firstly, query images are converted to grayscale and resized to $W1 \times H1$. Then, the entropy map of size $W1 \times H1$ of these images are used to extract regions-of-interest (ROI). A region in an image is defined as a $W2 \times H2$ image patch. Thus, these $W1 \times H1$ images with regions/patches of size $W2 \times H2$ each contains RN regions, where $RN=(H1/H2)\times(W1/W2)$. The goodness matrix R based on the entropy map as shown below is then generated. Element r_{ij} in R equals to 1 when entropy score e_{ij} is greater than or equal to the goodness threshold GT, which means this region will be selected for matching. Only G regions which is informative enough for matching will be remained after this step.

$$\boldsymbol{R} = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1j} \\ r_{21} & r_{22} & \cdots & r_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ r_{i1} & r_{i2} & \cdots & r_{ij} \end{bmatrix} \tag{1}$$

$$If\ e_{ij} \geq GT, e_{ij} = 1; else\ r_{ij} = 0.$$

A corresponding HOG-descriptor of depth $4 \times L$ will be calculated for each ROI next. The query image HOG-descriptor is now a two-dimensional matrix with dimensions $[G, 4 \times L]$. After that, standard matrix multiplication between query and reference descriptor matrix and max pooling for the generated matrix will be implemented. In the end, we can get a similarity score in the range of 0 to 1.

Then, the similarity of images can be used to generate the information gain. The feature information gain is defined in [19] to determine if the information contained by a sequence is enough after adding a new frame to the sequence. By comparing the similarity score between first and subsequent images with the threshold to determine if newly added frame have provided sufficient information gain. However, in this research, the threshold to determine whether two images are similar enough is the same for all the images. In other words, when the similarity score between first image and newly added image is smaller than a same constant number, they assumed enough information gain have been achieved. In fact, when a same group of images is comparing to two different images, their averages of similarity score will be different, which means a constant threshold can't guarantee the same information gain for all images. So, they also had to set a limit for the number of added comparing images before obtaining enough information gain when the threshold is not appropriate. To solve this problem, we initialize a subset with two adjacent images ref_1 and ref_2 to get a suitable threshold for each subset. The similarity score of these two images will be stored as S_{init} which is generally the highest score because of the sequential nature of database. Then, similarity score between ref_1 and following images ref_3, ref_4, … will be compared to S_{init} minus a Difference Threshold DT to determine the information gain. In addition, we also discovered a trend that as the image comparison continued the similarity score between initial image and current comparing image will descend obviously, but when the score suddenly rises comparing to last score, this implies that the current image has contained enough new descriptors which can randomly generate higher scores. In this case, we also consider enough information gain has been achieved. So, two new constraints which is now our new information threshold to evaluate information gain are shown below. If any of them has been satisfied, the sequence reaches sufficient information gain.

$$\begin{cases} S < S_{init} - DT \\ S > S_{last} \end{cases} \tag{2}$$

B. Segmentation of Reference Database

Similarity score between one query image and two similar (usually adjacent in database) reference images tend to be close. Therefore, if a set of images is similar enough, we can choose one image to represent the whole set. In this way, we can simplify the entire database into a set of represent images to achieve hierarchical retrieval like [13]. However, in our method the clustering procedure is not conducted through the database for all images. Instead, temporal consistency inside each subset is required. We divide the database subsequently from the first reference frame using information gain to generate sequence subsets. The selection of represent image for each subset is determined by the number of regions in each image from ROI extraction, which is mentioned in Section III-A. The image with most regions is the most informative and can better describe its whole set. Then, all these represent images in each subset are combined to form a represent set. The represent set's size is much smaller than the original database. Each element in this set leads to a subset with similar frames. A two-dimensional set that stores every frame's descriptor is also created accordingly. The algorithm of whole segmentation process can be found in Algorithm 1. The reason why segmenting database subsequently in temporal order instead of clustering most similar images through the whole database is because two adjacent reference images tend to have similar scores even if they are both wrong. Instead of blindly relying on scores, which can lead to all candidates falling into an incorrect location with many high score frames, our segmentation strategy provides more options of different locations in candidates selection.

C. Hierarchical Retrieval

The process of the single-frame retrieval is illustrated in this section. Firstly, the similarity score between the query image and all the images in the represent set will be calculated. The HoG descriptor computation and ROI extraction will be implemented for query image, and descriptors for reference image has been stored previously. Then, only frames with top-N scores will be remained. After that, all similarity scores between query image and images in N subsets which are represented by the N represent images will be computed. Every subset will generate a local best matching image. The image with the highest score in N images is the final best match in single frame retrieval scenario. As for sequence match, all the best N images and their indexes in database will be stored for

following steps.

D. Sequence Query List Generation

Sequence-based VPR algorithm is more robust compared to single-frame method because supplementary information is provided by frames before current frame. Generally, the longer sequence contains more information and results in better performance. However, information gain is the actual feature that strongly influences the performance, instead of sequence length. More exactly, the enhancing effect of a newly added frame in sequence is determined by its dissimilarity with the current frame. In other words, including two similar images in a sequence can't improve the retrieval performance compared to include only one of them. The procedure to create our sequence query list is like ConvSequential-SLAM [19], keep comparing new frames before current frame until the sequence information gain reaches the threshold. But only the frame at where the threshold is just reached will be added to the sequence list, and all the interval images between this image and current image will be ignored. After a new frame is added to the sequence list, the newly added frame will now be regarded as current frame to find the next frame being added to the list. In our method, new images will be added to the sequence list iteratively M-1 times to create a sequence list of size M.

E. Sequence Match

After all the manipulations mentioned above, we obtain a distance matrix D of size $M \times N$, where M is the length of the query list and N is the number of candidate reference images for each element in that list. The matrix D as shown below contains $M \times N$ similarity scores between query and reference images. Since the sequence retrieval is limited in this constant size matrix, we can achieve a $O(1)$ complexity in this sequence matching procedure. In addition, the indices of reference frames in the database are also stored for a later filtering process. Where, d_{ij} is the similarity score between i-th query image in sequence and j-th candidate reference image.

$$\boldsymbol{D} = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{m1} & d_{m2} & \cdots & d_{mn} \end{bmatrix} \tag{3}$$

Most of previous sequence based VPR algorithms evaluate sequence score based on trajectory velocity, which is varied between V_{min} and V_{max} in steps of V_{step}. The sequences of different velocities in the map will be evaluated to find the best match. This sequence matching strategy assumed that the velocity during query and reference sequence is constant, which makes it unreliable. The constant velocity brings fixed frame indices interval between consecutive images in sequences which is an extremely useful constraint to find sequences of images in searching space. While in our method indices interval between images is based on information gain, and the same distance between consecutive images is not needed. So, such assumption is avoided in our method. Instead, we use three constraints as shown below to generate a trajectory in the searching space for evaluation. Where R_k is the index of corresponding reference image in database for kth query image in sequence list. The first inequality requires query and reference images must follow the same temporal order. Because the frame indices interval between adjacent images in the query list and their matching reference images should be closed.

Algorithm 1. Segmentation of database

Require:
The reference database, *DB*
n^{th} images in *DB*, ref_n
Ensure: A two-dimensional set containing all the segmented
subsets, *subset*
$m = 0$
$n = 0$
while $n <$ *length*(*DB*) **do**
$flag = 0$
last_score = 1
subset[m][0] = ref_n
subset[m][1] = ref_{n+1}
init_score = *similarity_compute_*func(ref_n, ref_{n+1})
$k = 2$
while *flag* = 0 **do**
curr_score = *similarity_compute_func*(ref_n, ref_{n+k})
if *curr_score* < *init_score* **-** DT **or** *curr_score* > *last_score* **then**
flag = 1
$n = n + k$
$m = m + 1$
else
subset[*m*][*k*] = ref_{n+k}
$k = k + 1$
last_score = *curr_score*
end if
end while
end while
return *subset*

We define the distance between one image in sequence list and the image that is added after it into query sequence is one *IT*, because new image is added when sequence reaches the information threshold exactly once. Applying the same information threshold as creating query sequence list to all reference images in database twice, we can get the image that is two *IT* before every reference image. These images are gathered into the *RL* set. And RL_k is the index of corresponding image in *RL* for kth query image in sequence. Since the distance between two adjacent query images in sequence is one *IT*, their corresponding reference images should remain an approximate same distance as well. Therefore, the distance between two reference images must not exceed two *IT*, and this requires R_k to be larger than RL_{k+1}. $score_1$, $score_2$, …, $score_m$ represent the similarity score between each query frame and their candidates. After the two filtering steps mentioned above, the combination of query images with highest summed score will finally be selected as the best match sequence.

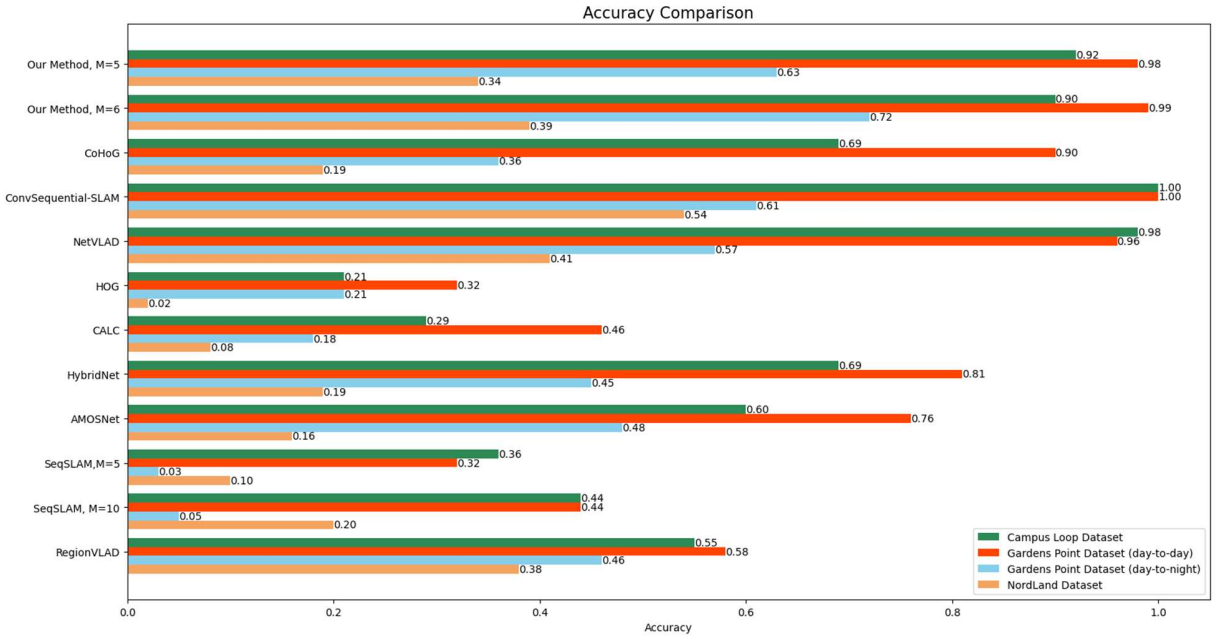

Fig. 2. Accuracy Comparison with other VPR algorithms

$$\begin{cases} R_1 < R_2 < \cdots < R_m \\ R_k > RL_{k+1} | k \in [1, m-1] \\ \underset{score}{argmax} \sum_{k=1}^{m} score_k \end{cases} \quad (4)$$

IV. Experimental Result

A. Datasets

To evaluate our proposed technique, the following public VPR datasets are used: Gardens Point dataset [23] containing images with viewpoint variation. This dataset consists of a total of 600 images, divided into 200 reference images (day images) and 400 query images, equally divided into day and night images. In this paper, we used day left as reference images, while day right and night right are used as query images. This dataset contains changes in images between day and night. The Nordland dataset [24] contains drastic appearance changes in different seasons. We have used the first 200 query images taken from the summer dataset and the first 200 reference images taken from the winter dataset. This dataset contains variations of images between different seasons. Campus Loop dataset [25] contains viewpoint variation, seasonal variation, and the presence of statically occluded frames. This dataset contains 100 queries and 100 reference images. All the datasets are applied to our method and other different methods in the same way. We have used the maximum size of these datasets, except for Nordland datasets which have too many images. Therefore, we choose 400 images from it of different seasons the same way as ConvSequential-SLAM. Although the number of images is limited, these images have contained sufficient changes in viewpoint, seasons, and day-night. So, they have been widely used in the VPR area by previous researchers.

B. Parameters

The parameters for descriptors extraction we used in this work are, $W1 = H1 = 512$, $W2 = H2 = 16$, $L = 8$ bins, $GT = 0.5$. As for the sequence match parameters, we set $DT = 0.03$, $M = 5$ and $M = 6$, $N = 20$. These values can provide a good performance in all the datasets we tested. Especially, the values of M and N could be set much smaller to obtain a faster online matching in a less challenging environment that didn't involve day-night or seasonal variation. For example, in the Gardens Point dataset (day-to-day), we set $M = 3$ and $N = 5$ and still achieve almost same accuracy as current M and N values. The influence of M and N on searching results will be well illustrated in Section IV-E and Section IV-F.

C. Database Segmentation

Firstly, we will calculate the similarity between images and decide if a sequence of adjacent images is similar enough to put in the same group. The mechanism explained in Section III.A and Section III.B is implemented to split the dataset into such groups. Then, for each group, we select one image with the most regions in ROI extraction to represent the whole group. Then, the size of searching space can be significantly reduced because the retrieval only needs to be done in the selected images of each group instead of the whole reference dataset.

D. Sequence Images Retrieval

After the reference dataset has been segmented, the query sequence can be generated and searched. Some generated query sequences and their matching results are shown in Fig. 3. These figures demonstrate that our sequence contains strong changes between adjacent images. Although the sequence is of length 6 in the figure, there are generally over 20 images between the first and last images in the database. Therefore, our method has managed to get a more informative sequence with a limited sequence length.

E. Accuracy Comparison

In our experiment, we use Intersection over Union (IoU) of two sequences to determine if two sequences are close enough to represent the same location. When IoU equals 0, this means

the trajectories of two sequences are totally irrelevant. And when IoU is larger than 0.5, this indicates the majority of two trajectories are overlapping at a common place. In this case, we consider two trajectories to be close enough and designate two sequences as a correct match. The IoU will be calculated according to frame indices in practical.

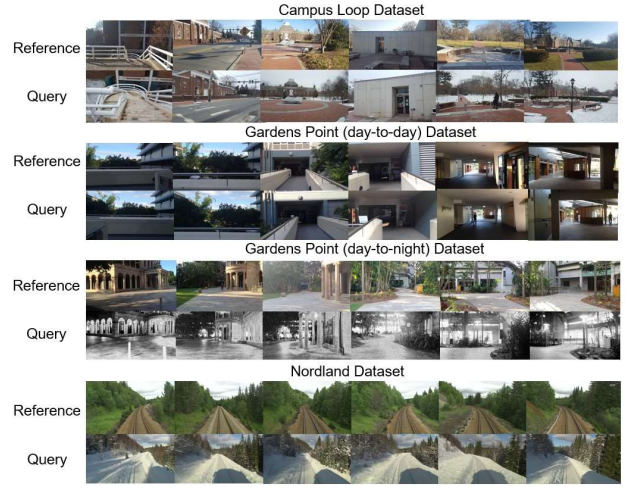

Fig. 3. Some correctly matched sequences of reference and query images

Then the accuracy of our algorithm has been compared with other previous research, such as ConvSequential-SLAM [19], NetVLAD [1], CoHOG [26], HOG [26], CALC [25], HybridNet [27], AMOSNet [27], SeqSLAM [6] and Region-VLAD [28] on the datasets mentioned in Section IV-A. The result is shown in Fig. 2. The parameter *M* in the graph is the sequence length. Our method has achieved comparable performance with other state-of-the-art algorithms like NetVLAD and ConvSequential-SLAM. Compared to our baseline method CoHoG, the performance has been improved significantly with supplementary sequential information. And our method outperformed all the other illustrated algorithms in the Gardens Point Dataset (day-to-night) which involves illumination variation. In addition, the results of our method with two different sequence lengths are illustrated to show the significant improvement in illumination and season variant dataset like Gardens Point dataset (day-to-night) and Nordland dataset caused by adding just one new frame in query sequence.

The performance of the proposed method is slightly inferior to ConvSequential-SLAM except on the point of the garden (day-to-night) dataset. This could be caused by the hierarchical search in our single-frame retrieval step. We only compared a small part of the database with each query image while ConvSequential-SLAM used the whole database. Hence, it is possible that the correct reference frame may not be found when it is not even compared with the query image. But the searching time for each frame can be reduced significantly. In addition, ConvSequential-SLAM used much longer and continuous query sequences, this is more time consuming, but it also contributes to a higher accuracy. In general, our method focuses more on computational efficiency so some compromise on accuracy is made.

F. Recall@k Evaluation

We have proposed a hierarchical single frame searching algorithm to achieve a faster retrieval speed compared to the baseline method CoHoG. The reduction of execution time will be illustrated in Section IV-F. Before that, it is necessary to prove that our method can remain the same accuracy performance. We will use recall@*k* as our performance metric to evaluate two methods in the illumination-variant Gardens Point dataset(day-to-night). Recall@*k* is defined as the ratio of correctly retrieved queries within the top *k* predictions to the total number of queries. Specifically, Recall@1 can represent the accuracy of single-frame retrieval. However, the performance when *k* increases is more important in our experiment because it is related to the selection of candidates for frames in query sequence. As shown in Table 1, two methods can achieve almost identically performance in single frame retrieval, and our method gradually exceeds CoHoG when *k* increases. This is because the segmentation of database helps us to filter out some high-score incorrect images clustering at false positive places.

Table 1. Recall@k of CoHoG and proposed method

	k=1	k=3	k=5	k=10	k=20
CoHoG	0.31	0.42	0.50	0.57	0.70
Ours	0.29	0.42	0.52	0.59	0.81

G. Various Sequence Length Performance

This section will show how the performance of our algorithm and ConvSequential-SLAM [19] varies because of changing sequence length. ConvSequential-SLAM is chosen because its descriptor extraction is also based on CoHoG, and it has managed to be one of the current state-of-the-art VPN algorithms. As shown in Fig. 3, the accuracy of two algorithms in Gardens Point dataset (day-to-day) and Campus Loop dataset which do not involve illumination variation is very close because the performance of single frame retrieval for images in query sequence is already fairly good. We will mainly focus on the other two challenging datasets. According to the graph, the accuracy of ConvSequential-SLAM improved slightly with each new adding frame in sequence. However, each new frame can significantly boost our algorithm accuracy since the information gain provided by the new frame is much larger than previous sequence-based algorithm. Moreover, the performance of our method at length 5 is comparable to static ConvSequential-SLAM at length of 20. Hence, our algorithm has proved to achieve a much more informative and compact sequence list for sequence match.

H. Computational Efficiency

Compared to models based on deep neural networks, the major advantage of handcraft features is the reduction of feature encoding time and RAM consumption. In our method, the CoHOG has been selected as the descriptor generator for each image. All the query and reference images are processed by CoHOG before the retrieval step. In Table 2, the comparison of computational efficiency between our feature encoder and other methods is illustrated. The comparison results in the table are obtained from [22] and [29]. According to the table, models based on deep neural networks like AlexNet and AMOSNet need more time and memory to encode the image. On the other hand, handcraft like HOG and CoHOG is much more efficient.

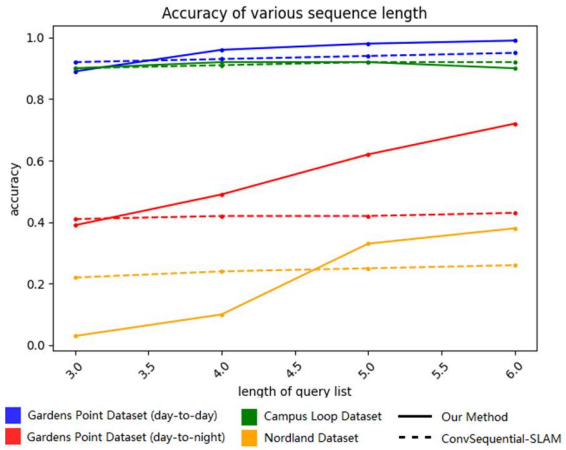

Fig. 4. Performance comparison under different sequence length

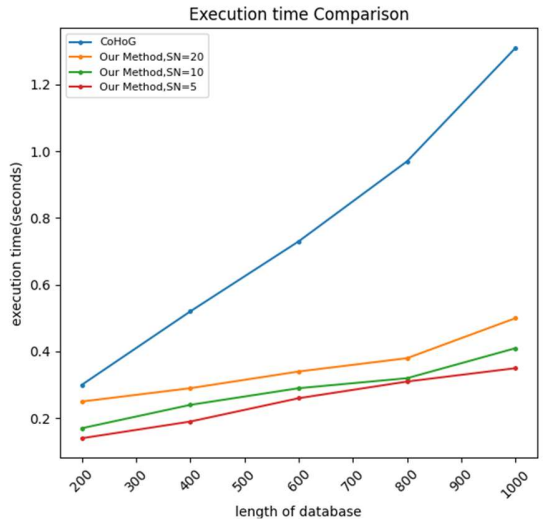

Fig. 5. Execution time comparison with CoHoG

The sequence-based VPR algorithms can be divided into two steps, single-frame candidates' selection for query frame in sequence and evaluation of sequences composed of selected candidates. In the first step, we used a hierarchical searching method to avoid an exhaustive search in the whole database like our baseline algorithm CoHoG. Figure 5 below shows the comparison of execution time between CoHoG and our algorithm with different SN values. SN is the selected number of candidate reference images. The search will only be carried out among SN candidates instead of the whole dataset. The length of datasets is the size of different partitioned Nordland datasets we used for experiments. Our method has much less execution time and smaller SN can further improve the performance. In the second step, the sequence length of our method is much smaller than previous sequence-based VPR algorithms and can still achieve a state-of-the-art-performance. This is because our query sequence is much more compact and contains more information. Obviously, the computational time is strongly affected by the sequence length. Therefore, the reduction of sequence length can lead to a more computationally efficient VPR algorithm. Moreover, velocity searching, which is very computationally expensive has been avoided during the sequence match procedure. Instead, three constraints are applied, as shown in Section III-E to evaluate sequences in searching space. Two filters also helped to reduce the number of sequences that needed to be evaluated in subsequent steps. Overall, benefitting from the strategies being applied in both steps, we managed to achieve a sequence-based algorithm with computational efficiency.

Table 2. Feature encoding time and RAM consumption per image comparison with other methods

	Encoding time (sec)	RAM consumption (MBs)
AlexNet	0.67	47.04
AMOSNet	0.36	4.22
HybridNet	0.36	4.33
CALC	0.03	2.30
NetVLAD	0.77	1.21
HOG	0.01	0.02
Ours	0.02	0.06

V. Conclusion

This paper presents a novel sequence-based searching algorithm for visual place recognition. Our algorithm managed to generate shorter and more compact query image sequences which helps to reduce the searching time. Compared to ConvSequential-SLAM, each newly added frame in the query sequence of our method can lead to more improvement in retrieval accuracy. Our method can maintain state-of-the-art accuracy with many short query sequences. The prediction accuracy of our method is over 0.9 in Gardens Point Dataset (day-to-day) and Campus Loop Dataset and 0.39 in Nordland Dataset which is only slightly inferior to the best methods in our comparison. And when query images contain illumination variation in Gardens Point Dataset (day-to-night), our accuracy reaches 0.72 which is the best of all methods. In addition, a hierarchical searching method based on the segmented database has been proposed to further improve computational efficiency. Our method's execution time managed to reduce to only 30% of CoHOG when the database's length is 1000. And the ratio can be further decreased if the size of the database is larger. In the future, an enhanced feature extraction strategy and a data expression method are further implemented to decrease the high storage capacity consumption for training.

References

[1] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR),* 2016, pp. 5297-5307.

[2] N. V. Keetha, M. Milford, and S. Garg, "A hierarchical dual model of environment-and place-specific utility for visual place recognition," *IEEE Robotics and Automation Letters,* vol. 6, no. 4, pp. 6969-6976, 2021.

[3] S. Hausler, S. Garg, M. Xu, M. Milford, and T. Fischer, "Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition." in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR),* 2021, pp. 14141-14152.

[4] M. Zaffar, A. Khaliq, S. Ehsan, M. Milford, & K. McDonald-Maier, "Levelling the playing field: A comprehensive comparison of visual place recognition

approaches under changing conditions," *arXiv preprint arXiv:1903.09107,* 2019.

[5] M. Zaffar, A. Khaliq, S. Ehsan, M. Milford, and K. McDonald-Maier, "Are state-of-the-art visual place recognition techniques any good for aerial robotics?" *arXiv preprint arXiv:1904.07967,* 2019.

[6] M. J. Milford, and G. F. Wyeth, "SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights," In *Proc. IEEE International Conference on Robotics and Automation,* 2012, pp. 1643-1649.

[7] M. Cummins, "Highly scalable appearance-only SLAM-FAB-MAP 2.0.," in *Proc. Robotics: Sciences and Systems (RSS),* 2009.

[8] Y. Hou, H. Zhang, and S. Zhou, "Tree-based indexing for real-time ConvNet landmark-based visual place recognition," *International Journal of Advanced Robotic Systems,* vol. 1, no. 14, 2017.

[9] D. Schlegel, and G. Grisetti, "HBST: A hamming distance embedding binary search tree for feature-based visual place recognition," *IEEE Robotics and Automation Letters,* vol. 4, no. 3, pp. 3741-3748, 2018.

[10] B. Harwood, and T. Drummond, T, "Fanng: Fast approximate nearest neighbour graphs." In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition,* 2016, pp. 5713-5722.

[11] M. G. Gollub, R. Dubé, H. Sommer, I. Gilitschenski, and R. Siegwart, "A partitioned approach for efficient graph–based place recognition," In *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst./Workshop Planning, Perception Navigat. Intell. Veh,* 2017, pp. 1-5.

[12] E. Garcia-Fidalgo, and A. Ortiz, "Hierarchical place recognition for topological mapping," *IEEE Transactions on Robotics,* vol. 33, no. 5, pp. 1061-1074, 2017.

[13] S. Garg and M. Milford, "Fast, compact and highly scalable visual place recognition through sequence-based matching of overloaded representations," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA),* 2020, pp. 3341-3348.

[14] A. Andoni and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," in *Proc. 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS-06),* 2006, pp. 459-468.

[15] Y. Weiss, A. Torralba, and R. Fergus, "Spectral hashing," in *Proc. Advances in Neural Information Processing Systems,* 2009, pp. 1753-1760.

[16] S. M. Siam and H. Zhang, "Fast-SeqSLAM: A Fast Appearance Based Place Recognition Algorithm," in *Proc. IEEE International Conference on Robotics and Automation,* 2017, pp. 5702-5708.

[17] P. Neubert, S. Schubert, and P. Protzel. "Exploiting intra database similarities for selection of place recognition candidates in changing environments," In *Proc. of the CVPR Workshop on Visual Place Recognition in Changing Environments,* 2015.

[18] K. A. Tsintotas, L. Bampis, and A. Gasteratos, "DOSeqSLAM: Dynamic on-line sequence based loop closure detection algorithm for SLAM," In *Proc. IEEE International Conference on Imaging Systems and Techniques (IST),* 2018, pp. 1-6.

[19] M. A. Tomită, M. Zaffar, M. J. Milford, K. D. McDonald-Maier, and S. Ehsan, "Convsequential-slam: A sequence-based, training-less visual place recognition technique for changing environments," *IEEE Access,* 9, pp. 118673-118683, 2021.

[20] S. Garg and M. Milford. "Seqnet: Learning descriptors for sequence-based hierarchical place recognition," *IEEE Robotics and Automation Letters,* vol. 6, no. 3, pp. 4305-4312, 2021.

[21] M. Chancán and M. Milford, "DeepSeqSLAM: A Trainable CNN+RNN for Joint Global Description and Sequence-based Place Recognition," *arXiv preprint arXiv:2011.08518,* 2020.

[22] M. Zaffar, S. Ehsan, M. Milford, and K. McDonald-Maier, "CoHOG: A light-weight, compute-efficient, and training-free visual place recognition technique for changing environments," *IEEE Robot. Autom. Lett.,* vol. 5, no. 2, pp. 1835-1842, Apr. 2020.

[23] N. Sünderhauf, S. Shirazi, F. Dayoub, B, Upcroft, and M. Milford, "On the performance of convnet features for place recognition," In *Proc. IEEE/RSJ international conference on intelligent robots and systems (IROS),* 2015, pp. 4297-4304.

[24] S. Skrede. (2013). Nordland Dataset. [Online]. Available: https://nrkbeta.no/2013/01/15/nordlandsbanen-minute-by-minuteseason-by-season/

[25] N. Merrill, and G. Huang, "Lightweight unsupervised deep loop closure," *arXiv preprint arXiv:1805.07703,* 2018.

[26] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc IEEE computer society conference on computer vision and pattern recognition,* 2005, vol. 1, pp. 886-893.

[27] Z. Chen, A. Jacobson, N. Snderhauf, B. Upcroft, L. Liu, C. Shen, I. Reid, and M. Milford, "Deep learning features at scale for visual place recognition," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA),* 2017, pp. 3223-3230.

[28] A. Khaliq, S. Ehsan, M. Milford, and K. McDonald-Maier, "CAMAL: Context-aware multi-layer attention framework for lightweight environment invariant visual place recognition," *arXiv preprint arXiv:1909.08153,* 2019.

[29] M. Zaffar, A. Khaliq, S. Ehsan, M. Milford, and K. McDonald-Maier, "Levelling the playing field: A comprehensive comparison of visual place recognition approaches under changing conditions," *arXiv preprint arXiv:1903.09107,* 2019.