2022 7th IEEE International Conference on
Advanced Robotics and Mechatronics
July 9-11, 2022, Guilin, China

2022 International Conference on Advanced Robotics and Mechatronics (ICARM) | 978-1-6654-8306-3/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICARM54641.2022.9959319

TLCD: A Transformer based Loop Closure Detection for Robotic Visual SLAM

Chenghao Li, Hongwei Ren, Minjie Bi, Chenchen Ding, Wenjie Li, Rumin Zhang, Xiaoguang Liu, Hao Yu[1]

***Abstract*— Loop closure detection (LCD) can effectively correct errors in visual odometry. It is thereby a critical part in robotic visual simultaneous localization and mapping (SLAM) system, which is widely used in modern robotic systems such as sweeping robots and drones. In this paper, we propose a transformer-based loop closure detection algorithm (TLCD), which employs a distillation transformer as backbone to extract global features, and is combined with a sequence matching as back-end processing of principal component analysis (PCA) algorithm. TLCD can accurately provide Precision-Recall curve based on several public datasets including CityCentre and New-College datasets. Results show that TLCD's average accuracy is up to 16.91% higher than the traditional LCD method. It is also about 3.18% higher accuracy than the state-of-the-art convolutional neural network (CNN) based LCD method.**

I. Introduction

With the continuous introduction of robots into people's life, the demand for intelligent robots such as sweeping robots and drones is becoming higher and higher. How to build a map of the environment and localize the agent within the environment have become a hot research field. Then SLAM [1] technology allows the simultaneous localization and navigation of robots without prior environmental information. The robot senses the environment and estimates its own position. Lidar SLAM technology has been very mature [2]. However the vision-based SLAM is still in the research stage and has extremely high research value.

After proposing the advanced visual odometry (VO) system named An Attentive Tensor-compressed LSTM Model with Optical Flow Features for Monocular Visual Odometry (ATFVO) [3], we started to focus on another important part of SLAM, loop closure detection (LCD) [4], which is mainly used to detect whether the robot returns to the position it once passed during the movement. The importance of LCD is to eliminate cumulative error. The cumulative error refers to the continuous accumulation of error generated when the visual odometer estimates the trajectory in SLAM, resulting in cumulative drift between the estimated trajectory and the actual trajectory. In order to make SLAM more real-time, the accuracy of the odometer is often sacrificed, leading to even higher accumulated.

Existing loop closure detection methods have been proved to be unable to achieve efficient and accurate prediction of the true closed-loop, as shown in Fig.1. At present, most LCD is carried out through the comparison of image feature similarity between frames. The traditional methods are mostly based on the appearance of the same scene, using artificial markers of scene image feature extraction of feature points. This method works well when environment appearances of the scene are constant , such as indoor scene. But for the environment of changeable outdoor scene, this method lacks certain robustness.

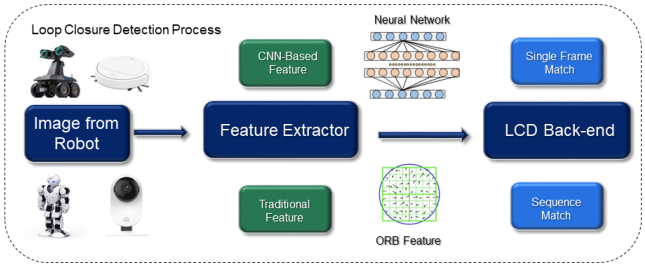

Fig. 1. Modules for different LCD algorithms

Recently, with the development of deep learning, LCD using deep learning method has become a popular trend. Convolutional neural network (CNN) [5] is the most widely used backbone. Specifically, pre-trained mode is trained to extract image features. This method can produce feature expression with more information, followed by improved average accuracy and robustness of the LCD.

Transformer [6] has changed the world of machine learning since it was introduced in the field of natural language processing (NLP). Subsequently, Dosovitskiy et al. [7] applied Transformer structure to computer vision and proposed Vision Transformer (ViT). It achieved excellent results on many datasets, compared with the most advanced convolutional neural networks at present. This shows the potential of Transformer to replace and surpass traditional CNN in computer vision.

In this paper, a vision transformer and sequence matching loop closure detection system called TLCD is proposed. To summarize, this paper makes the following four main contributions:

1) Feature extraction of RGB images with Transformer, generating image features required by the back end of loop closure detection.
2) Training transformer using Places365 [8] to get the weight of the scene classification.
3) Sequence matching is designed at the back end of loop

This work was supported by the National Natural Science Foundation of China (NSFC) (Key Program Grant No. 62034007), Innovative Team Program of Education Department of Guangdong Province (Grant No. 2018KCXTD028), and Shenzhen Science and Technology Program (Grant No. KQTD20200820113051096) and NSQKJJ (Grant No. K21799101).

[1]This work is with School of Microelectronics, College of Engineering, Southern University of Science and Technology, 1088 Xueyuan Avenue, Nanshan District, Shenzhen, Guangdong, China

978-1-6654-8306-3/22/$31.00 ©2022 IEEE

closure detection for improving the average accuracy.
4) The loop closure detection proposed by us achieves competitive average accuracy.

The structure of the article is as follows. Section II introduce related works in this filed. Section III introduces the framework of the proposed model, its details and mathematical principles. Section IV presents the relevant experiments and results, as well as our analysis of the results. Section V summarizes this work and discusses aspects of this work that motivate future research.

II. Related Work

In this section, we touched on some related works done by the predecessors. For loop closure detection, there are great differences between traditional methods and deep learning-based methods. Recently, Transformer has shown great potential in the visual field, and many researchers have proposed many different efficient models.

A. Loop Closure Detection

In traditional loop closure detection methods, the most commonly used algorithm is Bag-of-Words (BoW) [9]. This algorithm is based on a known word bag and obtains a dictionary of a certain size through k-means and other regression operations on a large number of data, in which each "Word" represents a kind of feature. The model is then used to characterize the image features. This approach has the advantage of being efficient in practice, but for large dictionary, storage becomes a problem. In the process of obtaining the dictionary, it is necessary to get the features from the existing images. Traditional methods mainly rely on artificial features to extract the information of scene images. Key point features are commonly used, such as Scale Invariant Feature Transform (SIFT) [10], which is a local feature descriptor with good stability and scale invariance. Speedup Robust Features (SURF) [11] is also a robust descriptor of local feature points. It is based on scale space and can maintain invariance for image scaling, rotation, and other transformations. The FAB-Map [12] algorithm proposed by Cummins et al. used SURF features and has achieved good results. However, due to the huge cost of computation, they are not suitable for real-time application, especially being applied to edge devices with limited computing capacity. In order to solve this problem, binary feature extraction algorithms emerged later. For example, the FAST Detector [13] method combined with Binary Robust Independent Elementary Features (BRIEF) [14] descriptor was proposed. FAST can quickly search out potential feature points in a graph. BRIEF is a feature descriptor. Although it does not have the features of rotation and scale invariance, its computational complexity is low, and it has high speed. To overcome its disadvantages, algorithms such as Oriented FAST and Rotated BRIEF (ORB) [15] and Binary Robust Invariant Scalable Keypoints (BRISK) [16] appeared.

In recent years, many methods based on deep learning have been proposed in computer vision. Because deep learning can learn deeper features of images, rather than traditional appearance features, it is more robust. Because of their outstanding performance in the above fields, they are also used in LCD.

First, researchers usually use the fully trained convolutional neural network (CNN) pre-training model to compare the extracted feature vectors and calculate the similarity matrix to get the similarity relationship between different frames, so as to judge whether it is a closed loop. Chen et al. [17] proposed a multi-scale deep feature fusion based LCD scheme. AlexNet [18], pre-trained on ImageNet [19], is also used as the feature extraction network. As proposed by Xia et al. [20], cascading deep learning network (PCANet) [21] is used to extract image features for comparison. Subsequently, Xia et al. [22] compared the LCD system based on traditional feature extraction with those based on CNN, including many famous models, such as PCANet [21], AlexNet [18], CaffeNet and GoogleNet [23]. The results show that the latter has higher average accuracy than the former when using the same test dataset, and has a great advantage in running time. However, the problems are also obvious. Firstly, the average accuracy of the system is not high, which can not meet the requirements of practical application. Secondly, in the face of some dynamic shielding objects (such as cars on the road, indoor pedestrians, etc.), its robustness is poor. In addition, when the environment changes (such as rain, night, etc.), the system will have worse detection effect.

The second is based on autoencoder and unsupervised learning. Visual loop closure detection based on autoencoder proposed by Merril et al. [24] is realized using unsupervised deep neural network. They add random noise to the training input and simulate the natural viewpoint change caused by robot motion by random projection transformation. They also used histogram of Oriented Gradients (HOG) [25] to generate illumination invariance and geometric features, and then had the encoder generate HOG descriptors. However, the autoencoder is unable to show which particular frame matches the current image. Instead, it only tells whether the current position has been accessed.

B. Self-Attention and Transformer in Vision

Transformer achieved significant improvements when it was first applied to NLP. Vaswani et al. [6] proposed Transformer based on attention mechanism for machine translation and English constituency parsing tasks. In addition to this, Transformer has shown revolutionary performance improvements in the CV space since the end of 2020. Vision Transformer (ViT) [7] is a model proposed by Dosovitskiy et al. that applies Transformer in image classification, and many subsequent works are improved based on ViT. The idea behind ViT is simple: the image is directly divided into patchs of fixed size, and then patch embedding is obtained by linear transformation, which is similar to NLP words and word embedding. Since Transformer input is a sequence of token embeddings, the patch embeddings of the image can be sent to Transformer for feature extraction and classification. Since ViT, the research on Vision Transformer has been in

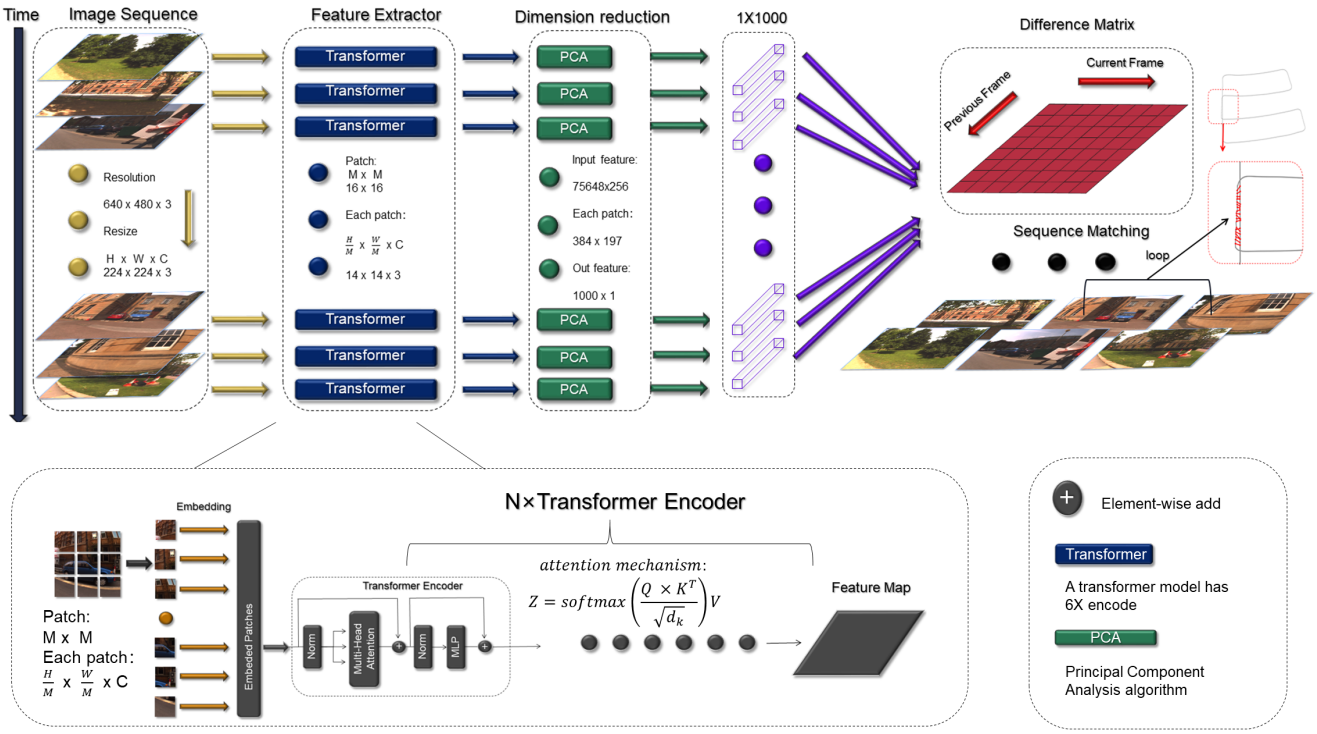

Fig. 2. Transformer-based LCD overall architecture

a blowout. Data-efficient image Transformers (DeiT) [26] was proposed to solve the problem of huge training data and the difficulty of setting hyperparameters in ViT. It uses data augmentation and distillation to significantly improve the performance of ViT model. Compared with ViT, Pyramid Vision Transformer (PVT) [27] introduces a pyramid structure similar to CNN, making PVT as backbone applied in dense prediction tasks. Finally, a new model named Swin Transformer [28], has broken records in many tasks, making Transformer structures become the new mainstream of vision.

III. TLCD Framework

In this section, we will elaborate on the framework of TLCD system in detail, and then clearly explain the principle of each part. It includes feature extractor and back-end sequence matching. The feature extraction part mainly adopts pre-trained visual Transformer model and dimensional reduction. We give up the traditional individual comparison for each frame, but create the sequence matching method to improve the system effect.

A. System Framework

The architecture of the whole system is shown in Fig.2. The camera carried by the robot keeps getting images of the current position. We need to judge whether there is a closed loop at this position. We resize each of the input raw images in the sequence to the size of 224×224×3. Each image is then fed into a feature extractor, during which the image is passed through a pre-trained Transformer model. For each image, an original feature map is generated. Because the number of parameters of the original feature vectors are too large, which affects the real-time performance of the system, we conduct principal component analysis (PCA) [29] on them and finally obtain the feature vector of each image, with the size 1×1000. Then we use these feature vectors to calculate the similarity according to the sequence and obtain the difference matrix, which corresponds to the similarity information between each image and each previous sequence. Finally, by artificially setting a certain threshold value, the difference matrix is used to judge whether there is a closed loop , and the position is obtained.

B. Transformer Based Feature Extractor

The feature extractor in the front end only aims to get a description vector of the image. The application environment of SLAM is generally some scene pictures, so we can naturally simplify the problem into using transformer to achieve scene classification, and then remove the classification header from the obtained model to obtain the feature representation of the scene image.

Transformer has superior ability to encode and decode words, and can be considered a more advanced bag-of-words model. LCD tasks based on traditional methods usually convert images to words, use words to represent an image, and then compare the image's similarity. The transformer method based on the advanced bag-of-words model is similar in principle to the traditional method. We found through experimental results that it has a encoding ability that is more

suitable for LCD tasks. The requirement of the LCD task is to compare whether the scenes at the two positions are consistent. The CNN-based algorithm tends to be consistent in each feature, and does not show obvious advantages in dynamic scenes. When the transformer algorithm is used in the case of occluding part of the picture, it can still show relatively good coding classification ability. Next, the flow and mathematical principles of the algorithm will be introduced in detail.

As shown in Fig.2, for an resized image $x \in R^{H(224)\times W(224)\times C(3)}$ from dataset, it is first divided into $M(16\times 16)$ patches with fixed size $L(14)\times L(14)$, namely $M=\frac{H\times W}{L^2}$. After considering the channel, the image is transformed into a list of input $x_L \in R^{M\times(L^2\times C)}$. A D dimensional class embedding is added to the list after embedding as shown in formula(1), which is used as input to transformer encoder after positional encoding. The transformer encoder module uses the original structure.

$$T_0 = [x_{class}; x_l^1 Emb; x_l^2 Emb; ...; x_l^N Emb] + Emb_{pos} \quad (1)$$

$$T_l' = MSA(LN(T_{l-1})) + T_{l-1} \quad (2)$$

$$T_l = MLP(LN(T_l')) + T_l' \quad (3)$$

$$y = PCA(LN(T_L^0)) \quad (4)$$

The attention mechanism relies on three trainable parameters of (query, key, value) to build. The input x is multiplied by the weight W^q to get a query vector $q \in R^t$. Then the query vector uses inner products to match against a list of key vectors $k \in R^k$, and the outputs $K \in R^{k\times d}$ are scaled and normalized by $\sqrt{d}$. The output of the final soft attention is the result of the previous step multiplied by $v \in R^{k\times d}$. The calculation process of the attention can be seen from formula(5).

$$Attention(Query, Key, Value) = Softmax(\frac{QK^T}{\sqrt{d}})V \quad (5)$$

Vaswani et al. [6] propose a self-attention layer. A sequence of input vectors X multiply weight to get three important parameter (query, key, value): $Q = XW^Q$, $K = XW^K$, $V = XW^V$. Finally, a multi-head self-attention (MSA) mechanism with different weight is formed through a single self-attention mechanism. The input apply h self-attention functions to get different outcome. Each head calculates the outcome by a sequence of $R^{M\times d}$. The final multi-head attention result is unified the dimension $M\times D$ from $M\times dh$ by the Forward Neural Network layer.

C. Sequence matching

Mature LCD often compares the similarity between feature vectors of single images one by one to judge whether it is closed loop, but this ignores very important sequence information. What we do is to use that information.

First of all, we need to get a difference matrix D, which is used to represent the similarity between each picture. Here, cosine similarity is used:

$$similarity = cos(\theta) = \frac{P_i \cdot P_j}{||P_i||\,||P_j||} \quad (6)$$

False Positive(FP) Sample

False Negative(FN) Sample

Fig. 3. Part of the LCD Dataset False Matching

TABLE I
CLASSIFICATION OF LOOP CLOSURE DETECTION RESULTS

P \ GT	Loop	Not Loop
Loop	True Positive (TP)	False Positive (FP)
Not Loop	False Negative (FN)	True Negative (TN)

$$D(i,j) = norm(1 - similarity) \quad (7)$$

where P_i and P_j respectively represent the feature vectors of the two images to be compared, and *norm* represents the normalization operation. In order to make the difference matrix more obvious, we also do local contrast enhancement.

For the current image $P_{current}$ to be detected, there may be a sequence matching with its sequence, but the length of the sequence is not fixed because the speed of the robot passing through the same place is different, so we set different speed (v_{max}-v_{min}) for matching. For each starting image we set in the database, we apply different speeds to it, generating a potential matching sequence. By calculating the total difference value of each sequence, the minimum total difference value of all speeds is finally obtained, which is the potential closed loop.

IV. EXPERIMENTS

In this section, we will clearly illustrate the effect of TLCD through a series of experiments. We'll start with the datasets we used, including training transformer and evaluating loop closure detection. Then we go through the details of how we train the transformer model on the scene dataset Places365, and the results of the training, and compare it with other CNN-based methods. Next, we combine the sequence matching algorithm to the trained Transformer model with achieve loop closure detection, evaluate its effect, compare with the state-of-the-arts algorithm and test the system average precision.

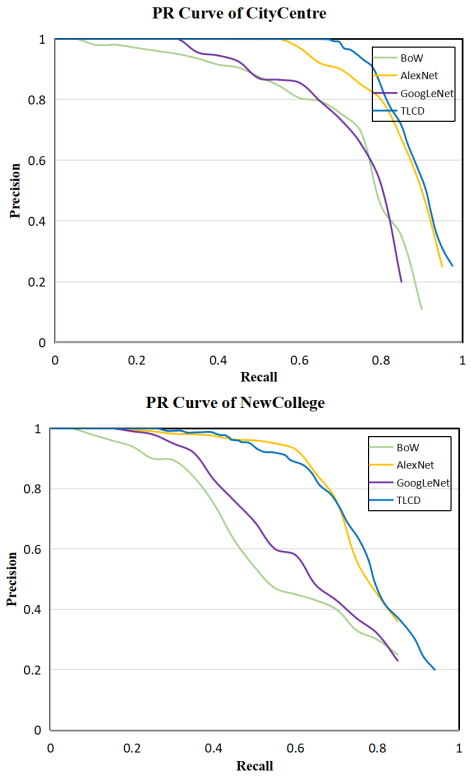

Fig. 4. PR curves of TLCD in CityCentre and NewCollege Compared with Other Methods

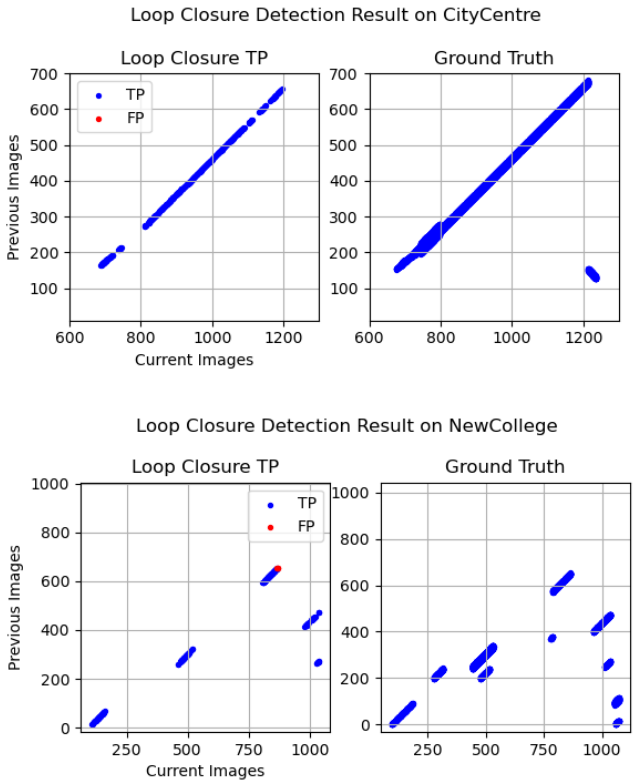

Fig. 5. Loop matrix of TLCD in CityCentre and NewCollege

A. Datasets

Places365-standard [8] is a large dataset created for tasks such as scene recognition. There are 365 categories, including indoor and outdoor scenes, containing 1.8 million images. Each category has 50 images for the validation set and 900 images for the testing set. In this paper, we will use this dataset to train the visual transformer.

In this paper, two public datasets, NewCollege and CityCentre [12], collected by Mobile Robotics Group of Oxford University, are used for test experiments. In these datasets, the robot walks along a certain route in an outdoor environment, and the data is collected by two cameras on both sides of the robot every 1.5 seconds. As the most authoritative datasets in the field of loop closure detection, it contains real closed-loop information and can effectively test the system performance. Among them, the dataset of NewCollege contains 2146 images, and the dataset of CityCentre contains 2474 images, each with a size of 640×480×3. In these two datasets, the groundtruth of the closed loop is provided to the users in the form of matrix GT. The two dimensions of the matrix are the same data index. If frame i and frame j ($i>j$) are closed loop, then the corresponding $GT(i,j)$ is 1; otherwise 0.

B. Training on Places365

Since the LCD dataset is too small to train a deep learning model from scratch, we need to train it on a large scene classification dataset based on the transformer, so that it has the ability to encode and classify scenes.

TABLE II
TRAINING RESULTS OF DIFFERENT MODELS ON PLACES365

Method	AlexNet	GoogLeNet	VGG	ResNet	Transformer
top1	53.31%	53.59%	55.19%	54.65%	53.28%
top5	82.75%	84.01%	85.01%	85.07%	84.04%

Transformer are more difficult to train than CNN-based networks, and the state-of-the-art performance on ImageNet networks is the result of pre-training on private datasets. The CNN-based network has official open source model weights on the Places365 dataset, so we take the distillation method to train our transformer model, using ResNet50 as the teacher network. The final training results are shown in TABLE II. Our model shows the same advantages as the CNN-based model in classification results, but from the back end, the features extracted by the transformer are more suitable for use in LCD scenes.

C. Loop Closure Detection Results

In loop closure detection, when it comes to evaluation, we use Precision-Recall Curve (P-R Curve) and average accuracy.

There are four different situations in LCD, as shown in TABLE I. False positive is when two images look similar but are not actually the same scene. False negative is when two images are actually the same scene, but it is not detected.

TABLE III
AVERAGE ACCURACY (AP) OF TLCD COMPARED WITH OTHER METHODS

Feature	Traditional		Transformer
	BoW	GIST	
NewCollege	62.38%	60.82%	**79.29%**
CityCentre	72.64%	69.79%	**89.05%**

CNN-Based				
PCANet	CaffeNet	AlexNet	GoogLeNet	VGG
73.76%	74.04%	78.92%	66.47%	78.26%
81.38%	82.80%	85.87%	74.19%	82.13%

The samples are shown in Fig.3. We want false positives and false negatives as few as possible, and the other two as many as possible. So we define Precision and Recall as:

$$Precision = \frac{TP}{TP+FP} \tag{8}$$

$$Recall = \frac{TP}{TP+FN} \tag{9}$$

It can be seen from the above formula that it is impossible for an algorithm's to have high Precision and Recall at the same time, so P-R curve is used to characterize the relationship between them. The larger the area enclosed by the generated curves, the better the algorithm performance. The area enclosed by the P-R curve can be defined as average precision (AP):

$$AP = \int_0^1 P(r)\,dr \tag{10}$$

where $P(r)$ is the function of the P-R Curve.

We resize each image in the datasets to 240×240×3 in sequence, then normalize and input into the pre-trained vision transformer model. The obtained feature map was reduced to 1×1000 feature vector by PCA. The difference matrix is obtained by calculating the cosine similarity of these vectors. Finally, by using sequence matching algorithm, we get the closed-loop prediction results.

In the experiment, we calculate a series of Precision and Recall combinations by setting different thresholds for the difference values, thus drawing the P-R Curve. At the same time, we tested other methods and drew them together, as shown in Fig.4. We find that the P-R curve of TLCD is far better than that of traditional methods, and also better than that of many CNN methods.

We find that for CityCentre, when recall value is around 60% and precision value is close to 100%, loop closure detection results are the most ideal. For NewCollege, results are the best when Recall is around 40% and Precision is close to 100%. The experimental results are shown in Fig.5, where the X-axis represents the image being processed and the Y-axis represents the image to be matched.

Finally, we calculate the AP of TLCD and compare it with those of traditional artificial features (BoW, GIST) and CNN features (PCANet, CaffeNet, AlexNet, GoogLeNet, VGG), as shown in TABLE III. In NewCollege, TLCD is 16.91% higher than traditional BoW and 18.47% higher than GIST. Compared with the CNN method, TLCD is 5.53% higher than PCANet, 5.25% higher than CaffeNet, 12.82% higher than GoogLeNet, 1.03% higher than VGG, and 0.37% higher than AlexNet. In CityCentre, TLCD is 16.41% higher than traditional BoW and 19.26% higher than GIST. Compared with the CNN method, TLCD is 7.67% higher than PCANet, 6.25% higher than CaffeNet, 3.18% higher than AlexNet, 14.86% higher than GoogLeNet, and 6.92% higher than VGG. In conclusion, the performance of TLCD is superior to existing LCD methods.

TABLE IV
COMPARATION OF THE MODEL SIZE BETWEEN DIFFERENT MODELS

Name	Structure	Size
AlexNet	CNN	>200MB
VGG	CNN	>500MB
ViT	Transformer	>300MB
TLCD	Distilled Transformer	86MB

TABLE V
TIME CONSUMPTION OF TLCD WHEN EVALUATE ON PC(CPU)(S/FRAME)

	CityCentre		NewCollege	
	Feature Extractor	Total LCD	Feature Extractor	Total LCD
TLCD	0.948	1.075	1.066	1.186

We also tested the time performance of TLCD. As the model size largely affects the running time and efficiency of the system, we first record the size of different deep learning models suitable for LCD systems. As shown in TABLE IV, the model size of TLCD is 86MB, which is generally lower than other models. We then test the system's time performance on two datasets mentioned above, expressed as the average time of each image in the entire dataset. We recorded the time of feature extractor and total LCD respectively, as shown in TABLE V.

V. CONCLUSION

This paper proposes a novel loop closure detection that employs an advanced Bag-of-Words transformer algorithm as the front-end, combined with sequence matching and dimension reduction algorithms as the back-end. This architecture has significant accuracy improvements on LCD evaluation metrics when conducting experiments on LCD datasets when compared to traditional methods. In the future work, we will apply tensor train methods [30], and perform neural architecture search [31] to optimize the above proposed method on edge devices.

REFERENCES

[1] M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (slam) problem," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, 2001.

[2] S. Arshad and G.-W. Kim, "Role of deep learning in loop closure detection for visual and lidar slam: A survey," *Sensors*, vol. 21, no. 4, p. 1243, 2021.

[3] H. Ren, C. Li, X. Zhang, C. Ding, C. Man, and H. Yu, "Atfvo: An attentive tensor-compressed lstm model with optical flow features for monocular visual odometry," in *2021 WRC Symposium on Advanced Robotics and Automation (WRC SARA)*, 2021, pp. 79–85.

[4] J. Fuentes-Pacheco, J. Ruiz-Ascencio, and J. M. Rendón-Mancha, "Visual simultaneous localization and mapping: a survey," *Artificial Intelligence Review*, vol. 43, no. 1, pp. 55–81, 2015.

[5] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[8] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, pp. 1452–1464, 2017.

[9] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2, 2006, pp. 2161–2168.

[10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[11] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[12] M. Cummins and P. Newman, "Fab-map: Probabilistic localization and mapping in the space of appearance," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 647–665, 2008.

[13] D. Galvez-Lopez and J. D. Tardos, "Real-time loop detection with xbags of binary words," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 51–58.

[14] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," in *European Conference on Computer Vision*, 2010, pp. 778–792.

[15] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International Conference on Computer Vision*, 2011, pp. 2564–2571.

[16] S. Leutenegger, M. Chli, and R. Y. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *2011 International Conference on Computer Vision*, 2011, pp. 2548–2555.

[17] B. Chen, D. Yuan, C. Liu, and Q. Wu, "Loop closure detection based on multi-scale deep feature fusion," *Applied Sciences*, vol. 9, no. 6, p. 1120, 2019.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.

[19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[20] Y. Xia, J. Li, L. Qi, and H. Fan, "Loop closure detection for visual slam using pcanet features," in *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, pp. 2274–2281.

[21] T.-H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "Pcanet: A simple deep learning baseline for image classification?" *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5017–5032, 2015.

[22] Y. Xia, J. Li, L. Qi, H. Yu, and J. Dong, "An evaluation of deep learning in loop closure detection for visual slam," in *2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2017, pp. 85–91.

[23] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

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

[25] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, 2005, pp. 886–893.

[26] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *International Conference on Machine Learning*, 2021, pp. 10 347–10 357.

[27] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," *arXiv preprint arXiv:2102.12122*, 2021.

[28] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," *arXiv preprint arXiv:2103.14030*, 2021.

[29] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[30] Y. Cheng, G. Li, N. Wong, H.-B. Chen, and H. Yu, "Deepeye: A deeply tensor-compressed neural network for video comprehension on terminal devices," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 19, no. 3, pp. 1–25, 2020.

[31] T. Elsken, J. H. Metzen, and F. Hutter, "Neural architecture search: A survey," *The Journal of Machine Learning Research*, vol. 20, no. 1, pp. 1997–2017, 2019.