Detection of loop closure in visual SLAM: a stacked assorted auto-encoder based approach*

LUO Yuan (罗元)[1], **XIAO Yuting** (肖雨婷)[1]**, **ZHANG Yi** (张毅)[2], **and ZENG Nianwen** (曾念文)[1]

1. Key Laboratory of Optoelectronic Information Sensing and Technology, School of Optical Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

2. Engineering Research Center for Information Accessibility and Service Robots, School of Advanced Manufacturing Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

(Received 14 October 2020; Revised 30 October 2020)

The current mainstream methods of loop closure detection in visual simultaneous localization and mapping (SLAM) are based on bag-of-words (BoW). However, traditional BoW-based approaches are strongly affected by changes in the appearance of the scene, which leads to poor robustness and low precision. In order to improve the precision and robustness of loop closure detection, a novel approach based on stacked assorted auto-encoder (SAAE) is proposed. The traditional stacked auto-encoder is made up of multiple layers of the same autoencoder. Compared with the visual BoW model, although it can better extract the features of the scene image, the output feature dimension is high. The proposed SAAE is composed of multiple layers of denoising auto-encoder, convolutional auto-encoder and sparse auto-encoder, it uses denoising auto-encoder to improve the robustness of image features, convolutional auto-encoder to preserve the spatial information of the image, and sparse auto-encoder to reduce the dimensionality of image features. It is capable of extracting low to high dimensional features of the scene image and preserving the spatial local characteristics of the image, which makes the output features more robust. The performance of SAAE is evaluated by a comparison study using data from new college dataset and city centre dataset. The methodology proposed in this paper can effectively improve the precision and robustness of loop closure detection in visual SLAM.

Document code: A **Article ID:** 1673-1905(2021)06-0354-7

DOI https://doi.org/10.1007/s11801-021-0156-9

Simultaneous localization and mapping (SLAM)[1] refers to real-time localization and construction of a quantitative map of the environment as the robot moves through the unknown environment. Visual SLAM is capable of building three-dimensional (3D) maps of the environment in real time by using the camera as a sensor. A complete visual SLAM system consists of four modules: visual odometry, optimization, loop closure detection, and mapping[2]. Loop closure detection is a key module in visual SLAM, which plays a significant role in eliminating accumulated errors.

Loop closure detection is to determine whether the robot has returned to a certain position in the map when the current observation information and map information are given[3]. It is determined to be a loop closure when the similarity between the scene image at current location of the mobile robot and the previously visited scene image is greater than a set threshold. The most frequently used method is bag-of-words (BoW)[4]. The fast appearance based mapping 2.0 (Fab-Map 2.0) algorithm proposed by Cummins et al[5] used the external data image features of the explored area to form a BoW, and judged whether a loop closure is formed through comparing the vector probabilities generated by the two locations. To further improve the speed, DBoW2 (distributed bag-of-words, DBoW) was proposed, and Galvez-Lopez et al[6] built vocabulary trees to discretize the binary description space, a hierarchical structure that makes vocabulary lookups more convenient. Garcia-Fidalgo et al[7] introduced incremental bag-of-words loop closure detection (iBoW-LCD). The presented approach made use of an incremental bag-of-words (iBoW) scheme based on binary descriptors to retrieve previously seen similar images, avoiding any vocabulary training stage usually required by classic BoW models. Liu et al[8] took advantage of the global feature descriptor for loop closure detection. Zhang et al[9] applied the perspective invariant binary feature descriptors of images to an iBoW. Bampis

* This work has been supported by the National Natural Science Foundations of China (No.51905065), the Youth Program of National Natural Science Foundation of China (No.61703067), the Science and Technology Planning Project of Changshou District in Chongqing, China (No.CS2020007), and the Technology Innovation and Application Demonstration Project of Science and Technology Bureau of Beibei District in Chongqing of China (No.2020-5).

** E-mail: xiaoyt_cqupt@163.com

et al[10] segmented the image into sequences according to the motion trajectory, generating a description vector that can describe the overall characteristics of the scene for loop closure detection. G. Zhang et al[11] proposed a vocabulary construction algorithm called hierarchical sequence information bottleneck (HSIB) based on mutual information maximization mechanism (MMI), which enhanced the performance of loop closure detection algorithm. Liu et al[12] presented a loop closure detection algorithm based on CNN words. In this method, the elements of feature maps from the higher layer of the CNN are clustered to generate CNN words (CNNW). It inherits the characteristics of both CNN features and BoW methods. The original global description is transformed into a local description, and the data noise resistance is enhanced. Azam et al[13] proposed a novel approach using supervised and unsupervised deep neural networks based on super dictionary that does not need to generate vocabulary, which makes it memory efficient and instead it stores exact features. The supervised learning technique helped to avoid mobile objects to reduce the risk of false correspondence, whereas unsupervised learning technique is used to detect the possibility of loop closure to make it faster to process frames. The results proved that it could effectively detect loop closures even from slightly different viewpoints and in the presence of occlusions. However, BoW is very sensitive to environmental changes due to artificial-based design. It cannot provide a robust image feature description in actual scenes, resulting in a significant reduction in the precision of loop closure detection.

The combination of deep learning and loop closure detection mainly focuses on using deep neural networks to generate better descriptions of scene images. Gomez Ojeda R et al[14] trained AlexNet with Places dataset to extract features for loop closure detection. Gao et al[15] presented a loop closure detection algorithm based on stacked denoising auto-encoder (SDA), which obtained better results than the Fab-Map 2.0 algorithm proposed by Cummins et al[5]. However, the model extracts feature with high dimensionality and does not take into account the spatial local characteristics of images, which cause a poor robustness. Merrill N et al[16] devised a loop closure detection algorithm called convolutional auto-encoder for loop closure (CALC) based on a convolutional auto-encoder, in which the coding layer of the trained model is used as a feature extractor for visual SLAM keyframes. Although it performs well in datasets with drastic changes in illumination and viewing angle, the precision is limited. Burguera A et al[17] presented a Neural Network based on an autoencoder architecture, in which the decoder part is being replaced by three fully connected layers, aiming at robust and fast visual loop detection in underwater environments. Wang et al[18] proposed the stacked convolutional and autoencoder neural networks (SCANN) model for loop closure detection in visual SLAM based on the basic structure of convolutional neural networks. Image features extracted using the SCANN model have multiple invariances in image transformation. These features are more suitable for complex and varied real-world environments. Aritra et al[19] proposed a 12-layer deconvolution net that encodes and decodes an image to itself to learn the representation. The use of locally connected autoencoders in the network drastically reduces the dimension without significant loss in retaining the contextual information. Chen et al[20] developed a loop closure detection algorithm based on multi-scale deep feature fusion, which has strong robustness to environmental changes. However, the model needs to be trained using datasets containing a large number of labels, which is very difficult in practice.

In order to improve the precision and robustness of loop closure detection, a novel approach based on stacked assorted auto-encoder (SAAE) is proposed. The proposed SAAE is composed of multiple layers of denoising auto-encoder, convolutional auto-encoder and sparse auto-encoder. It is able to extract the features of the scene image and then output the features for loop closure detection. This unsupervised learning-based network model performs well in terms of generalization capability and robustness, and the dataset used for training does not need to carry labels, reducing the manual labeling effort.

Suppose there are two keyframes f_i and f_j, each keyframe can be expressed as t feature vectors:

$$f_n = \left\{v_1^{(n)}, v_2^{(n)}, \cdots, v_t^{(n)}\right\}, n = i, j\ . \tag{1}$$

Define a similarity function δ, using cosine distance to measure similarity between feature vectors, expressed as

$$s = \delta(f^{(i)} - f^{(j)}) = \frac{\sum_{k=1}^{t} v_k^i v_k^j}{\sqrt{\sum_{k=1}^{t}(v_k^i)^2}\sqrt{\sum_{k=1}^{t}(v_k^j)^2}}\ . \tag{2}$$

In loop closure detection, the similarity threshold for images is selected according to the following rules: taking a prior similarity $s(f_t, f_{t-\Delta t})$, it represents the similarity of the keyframe image at a certain moment to the keyframe at the previous moment. Other scores are normalized with reference to this value:

$$s(f_{t_i}, f_{t_j})' = \frac{s(f_{t_i}, f_{t_j})}{s(f_t, f_{t-\Delta t})}\ . \tag{3}$$

If the similarity of the current frame to a previous keyframe is more than three times to the last keyframe, it is considered that there may be a loop closure. The purpose of this method is to avoid introducing absolute similarity threshold, so that the algorithm can adapt to more environments[21].

The keyframe feature vector V_k is extracted by the proposed stacked assorted auto-encoder (SAAE), and its similarity score is calculated with the historical keyframe feature vector V_1, V_2, …, V_N. If it is greater than the set

threshold, it is judged to be a loop closure. The process is shown in Fig.1.

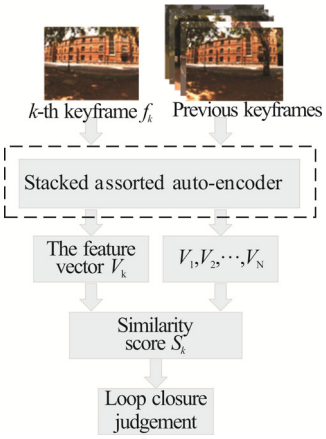

Fig.1 Loop closure detection process

Auto-encoder consists of input layer (x), hidden layer (h), and output layer (y)[22]. It reconstructs the input data by encoding and decoding, obtaining the hidden layer representation of the input data, so as to achieve the purpose of feature extraction. The mapping from the input layer (x) to the hidden layer (h) of the encoder is called encoding, which can be expressed as

$$h=f_\theta(x)=\sigma(\omega x+b). \tag{4}$$

The mapping between the hidden layer (h) and the output layer (y) is called decoding, as shown in the following formula:

$$y=g_{\theta'}(h)=\sigma(\omega' h+b'). \tag{5}$$

An error function is defined. By adjusting the parameters, the error between input sample and the reconstruction result converges to a minimum. The expression is as follows:

$$L(x,\ y)=\|x-y\|^2. \tag{6}$$

Denoising auto-encoder (DAE) is a variant of auto-encoder[23], which has better robustness and generalization ability. The structure of denoising auto-encoder is shown in Fig.2. The artificial input contains noise, and the clean input signal is reconstructed through the hidden layer.

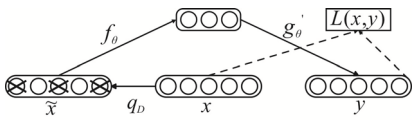

Fig.2 The structure of denoising auto-encoder

Convolutional auto-encoder (CAE) uses convolutional layers and pooling layers to replace the original fully connected layers, which has the characteristics of local connection and weight sharing. It can well retain the spatial local characteristics of images and be used as a hierarchical unsupervised feature extractor that adapts well to high-dimensional inputs[24]. The structure of convolutional auto-encoder is shown in Fig.3.

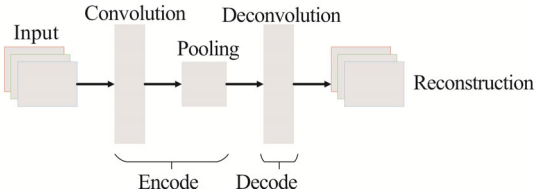

Fig.3 The structure of convolutional auto-encoder

Generally, the number of hidden layer nodes of the auto-encoder is set to be smaller than the number of input layer nodes to reduce the dimensionality of the input signal. In order to learn high-dimensional sparse features, the sparsity constraint is added to the hidden layer nodes, then a sparse auto-encoder (SAE) is obtained[25]. By suppressing most of the output of hidden layer neurons, the network achieves a sparse effect. On the basis of ensuring the precision of model reconstruction, it improves the performance of the model by greatly reducing the data dimension[26].

Sparse auto-encoder achieves inhibition by constraining the average activation value of the hidden layer neuron output, using Kullback-Leibler (KL) divergence[27] to force it to approximate a given sparse value, and adding it as a penalty term to the loss function.

Stacked auto-encoder[28], which is a neural network composed of multiple auto-encoders. The output of the former auto-encoder is used as the input of the latter auto-encoder. Stacked auto-encoder outperforms single auto-encoder by extracting deep features of images. Traditional stacked auto-encoder is often built with multiple layers of same kind of auto-encoder, a network that can easily lose features or create dimensional explosion problem. For the purpose of better extracting image features and further improving the robustness and generalization ability of the network model, an SAAE is designed in this paper to stack multiple auto-encoders, which can well combine the advantages of various auto-encoders to obtain a better network model for extracting features of images.

The proposed SAAE is composed of multiple layers of denoising auto-encoder, convolutional auto-encoder and sparse auto-encoder. The denoising auto-encoder improves the robustness of the network by artificially adding noise to the input signal. The features extracted by the hidden layer contain all the features of input images, and the original images can be reconstructed from the partially occluded or damaged images. The convolutional auto-encoder reduces the number of parameters by sharing weights, which simplifies the training process and well preserve the spatial information of images. The sparse auto-encoder is capable of extracting sparse features of input images, enabling dimensionality reduction while maintaining reconstruction precision. The specific network structure is shown in Fig.4.

In this paper, layer-by-layer training is used to train the model. Firstly, random noise is added to the training sample as input to a denoising auto-encoder, which is encoded to learn the low-dimensional features of images.

The reconstruction error is continuously reduced by using gradient descent. When the error reaches a minimum, it indicates that the training of the denoising auto-encoder is complete. Then, the output layer of the denoising auto-encoder is removed and its hidden layer is used as input for training the convolutional auto-encoder. The coding part of the convolutional auto-encoder designed in this paper consists mainly of four convolutional layers, all of which use small-sized convolution kernels to extract depth features, while ensuring the size of the local receptive field and reducing the parameters of the model. The decoding part consists of a three-layer fully connected network. After the original image is passed through the denoising auto-encoder and the convolutional auto-encoder, low-dimensional to high-dimensional feature extraction can be done layer by layer. Finally, the coding layer of the convolutional auto-encoder is used as input, with the sparsity constraint added to train the sparse auto-encoder, which reduces the dimensionality while extracting the abstract features of the image. The training hyper-parameters of SAAE designed in this paper are shown in Tab.1.

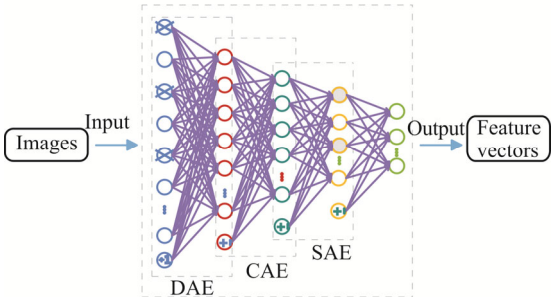

Fig.4 The structure of SAAE

Tab.1 Hyper-parameters in SAAE

Parameter	Value
Learning rate	0.01
Training epochs	500
Noise addition rate	0.15
Sparse coefficient	0.005

Stochastic gradient descent (SGD)[29] is used to iterate the network parameters. Fig.5 shows the training results at different learning rates. When the learning rate is set to 1.0, the error value always fluctuates around the initial value and cannot converge; when the learning rate is set to 0.5, the reconstruction error has a significant downward trend during the first 50 training epochs, and then keeps a small fluctuation, but fails to converge to the expected minimum; when the learning rate is set to 0.01 and 0.1, the reconstruction error curves exhibit nearly identical decreasing and converging trends, but the curve convergence trend corresponding to the learning rate of 0.01 is smoother. Therefore, this experiment set the learning rate at 0.01 and the number of training epochs 500.

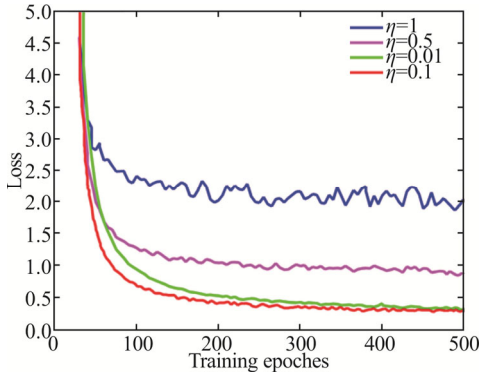

Fig.5 The training results at different learning rates

To train the denoising auto-encoder, random noise is added to the training sample, the result $\tilde{x}$ is input to the input layer, random noise v obeys a normal distribution with a mean of 0 and variance of σ^2. The noise addition rate is set to 0.15.

$$\tilde{x} = x + vx . \tag{7}$$

To train the sparse auto-encoder, the KL divergence is added to the loss function as a regular term to constrain the sparsity of the network. The loss function can be written as

$$J_{SAE}(w) = \sum(L(x,y)) + \beta\sum_{j=1}^{h} KL(\rho \,\|\, \hat{\rho}_j) , \tag{8}$$

$$\sum_{j=1}^{h} KL(\rho \,\|\, \hat{\rho}_j) = \sum_{j=1}^{h}(\rho \log\frac{\rho}{\hat{\rho}_j} + (1-\rho)\log\frac{1-\rho}{1-\hat{\rho}_j}) , \tag{9}$$

$$\hat{\rho}_j = \frac{1}{m}\sum_{i=1}^{m}(a_j(x_i)) , \tag{10}$$

where β is the weight of the sparse penalty term, which can take any value between 0 and 1. $\hat{\rho}_j$ is the training sample on the hidden layer neuron j of the average activation value, and a_j is the activation value on neuron j of the hidden layer. To achieve the effect that most of the neurons are inhibited, the sparse coefficient ρ generally takes a value close to 0. In this experiment, the sparse coefficient ρ is set to 0.005.

The convolutional auto-encoder involves many parameters, so they are listed separately, as shown in Tab.2. The pre-processed 160×120 images are used as input for training the convolutional auto-encoder. The coding part mainly includes four convolutional layers. In each layer of convolution operation, the convolution kernel moves along the x-axis and y-axis of the image. The first convolution layer has a convolution kernel size of 5×5, number of 32, stride of 1, padding of 2, pooling kernel size of 3×3, pooling stride of 2, using a local connection method, connecting 5×5 regions at a time, generating a 32×79×59 feature map after convolution and pooling. Subsequent calculations can be deduced by analogy. The second and third convolutional layers have identical parameters. The fourth convolutional layer has a convolutional kernel size of 3×3, number of 8, and stride of 1. After convolution and pooling, an 8×17×12 feature map can be generated. In the process of generating feature

maps, a weight-sharing strategy is adopted; in the process of moving the convolution kernel, the local field of view is gradually expanded to achieve the same effect as the full connection, while effectively reducing the number of parameters and saving computing space.

Tab.2 Parameters in CAE

Parameter	Conv 1	Conv 2	Conv 3	Conv 4
Conv kernel size	5×5	3×3	3×3	3×3
Kernel numbers	32	64	64	8
Conv stride	1	1	1	1
Pool kernel size	3×3	3×3	3×3	-
Pool stride	2	2	2	-
Padding	2	1	1	-

New college and city centre datasets[30] are standard datasets for evaluating the performance of visual SLAM loop closure detection algorithms. These two datasets are collected by a mobile robot platform carrying two cameras (one on each side), by which the distance traveled is 1.9 km and 2.0 km respectively. Images were collected approximately every 1.5 m of movement. The details of datasets are shown in Tab.3. These two datasets have manually calibrated ground truth loop closure, which can be used to compare with the loop closure obtained by the algorithm, so as to calculate the precision and recall.

Tab.3 The details of datasets

Dataset	New college	City centre
Number (frame)	2 146(1 073 pairs)	2 474(1 237 pairs)
Image size	640×480	640×480
Frequency	0.5 Hz	0.5 Hz
Image type	RGB	RGB
Description	outdoor	outdoor
Weather	sunny	sunny
Year	2008	2008
Sensor	Two cameras	Two cameras

The loop closure detection algorithm may be tested on a dataset with the four cases[31] shown in Tab.4.

Tab.4 Classification of loop closure detection results

Fact / Detection	True	False
Positive	True positive (TP)	False positive (FP)
Negative	False negative (FN)	True negative (TN)

In this paper, precision and recall are used to evaluate the performance of the algorithm. The specific representation is as follows:

$$Precision = \frac{TP}{TP + FP}, \tag{11}$$

$$Recall = \frac{TP}{TP + FN}. \tag{12}$$

In visual SLAM, there is a higher demand for precision, and good algorithms can still guarantee good precision at higher recall.

In this paper, a deep learning server is used to train an SAAE. Places365-Standard dataset[32] is used for training model, which is a scene-centric dataset. It has 1 803 460 training images with the image number per class varying from 3 068 to 5 000. Before training begins, each image in the training image set is converted to grayscale, resized to 120×160, then the training image pair obtained through the viewpoint variations is used as input. The well-trained model is used to learn the input images, and the output features are used for loop closure detection. Algorithm performance is tested on the new college and city centre datasets. The deep learning server configurations used for the experimental simulations are shown in Tab.5. Considering that the image size of the datasets used in this paper is 640×480 of which the feature dimension is too high, the image size was reduced to 160×120 by preprocessing. Use the trained network model to extract the feature vector of the image to calculate the similarity between the images. If it is greater than the set threshold, it is judged as a loop closure. The partial loop closure detected by this algorithm is shown in Fig.6. The image pairs with serial No.2 and No.96 in the new college dataset form a loop closure, and image pairs with serial No.158 and No.681 in the city centre dataset form a loop closure.

Tab.5 The configuration of deep learning server

Configuration	Details
CPU	Intel Xeon E5-2665, 2.4 GHz
RAM	16G
GPU	NVIDIA GeForce GTX1080Ti
System	Ubuntu 16.04 LTS
Development environment	TensorFlow1.8, Python3.7

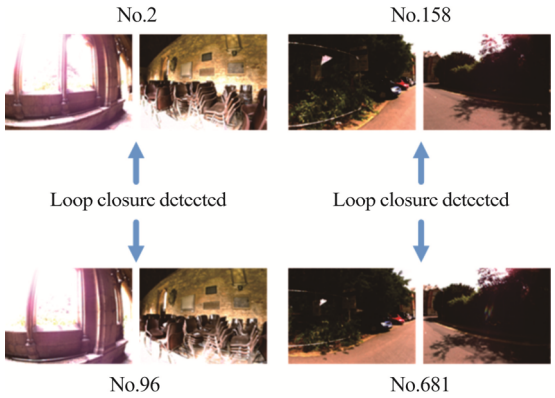

Fig.6 The partial loop closure detected by proposed algorithm

The proposed methodology has compared the performance with a classic bag-of-words based approach namely Fab-Map[5], a distributed bag-of-words based

approach namely DBoW2[6], an iBoW based approach namely iBoW-LCD[7], a stacked denoising auto-encoder based approach namely SDA[15] and a convolutional auto-encoder based approach namely CALC[16]. To compare with other methods, precision-recall curves have been generated for all, as shown in Fig.7.

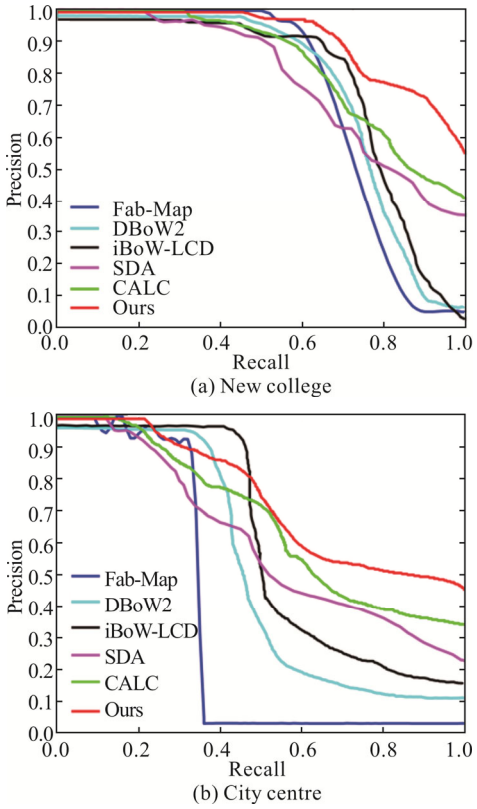

Fig.7 The precision-recall curves of the six methods

The precision of each algorithm on the new college and city centre datasets is shown in Tab.6 when the recall for loop closure detection is 80%.

Tab.6 Corresponding precision at 80% recall

Dataset	Method	Precision
New college	Fab-Map	23.8%
	DBoW2	36.2%
	iBoW-LCD	45.9%
	SDA	50.7%
	CALC	59.6%
	Ours (SAAE)	77.6%
City centre	Fab-Map	3.1%
	DBoW2	12.3%
	iBoW-LCD	21.1%
	SDA	36.3%
	CALC	39.2%
	Ours (SAAE)	51.2%

It can be seen from the chart that the average precision of the algorithm in this paper is better than the three algorithms of Fab-Map, DBoW2, iBoW-LCD, SDA, and CALC. When the recall of loop closure detection reaches 80%, the precision of three bag-of-words based approaches (Fab-Map, DBoW2 and iBoW-LCD) on the new college dataset respectively are 23.8%, 36.2% and 45.9%, and the precision of the algorithm in this paper is 77.6%. Although the precision of SDA and CALC is improved compared with three bag-of-words based approaches, it is still lower than the algorithm in this paper. On the city centre dataset, the performance decreases overall. When the recall of loop closure detection reaches 80%, the precision of three bag-of-words based approaches (Fab-Map, DBoW2 and iBoW-LCD) are only 3.1%, 12.3%, 21.1%. While the precision of the proposed algorithm is 51.2%, which is still higher than the three comparison algorithms. Thus, it can be seen that the algorithm in this paper is still able to guarantee good precision at relatively high recall.

To evaluate the temporal performance of the algorithm, the mean feature extraction time and loop closure query time are calculated on a sequence of keyframes from the new college and city centre datasets, as shown in Tab.7.

Tab.7 Mean time to extract features and query

Method	Mean time to extract features (ms)	Mean time to query (μs)
Fab-Map	234.12	196.35
DBoW2	19.58	173.25
iBoW-LCD	315.24	147.20
SDA	12.13	103.95
CALC	9.93	59.25
Ours (SAAE)	15.02	51.98

Fab-Map, DBoW2 and iBoW-LCD are loop closure detection algorithms based on bag-of-words, which is implemented based on CPU, and the algorithm in this paper, SDA, and CALC are all implemented based on GPU. The network of SDA is stacked only by denoising auto-encoders, while CALC builds a deep convolutional auto-encoder model incorporating HOG (Histogram of Oriented Gradient) features. Compared with the above two algorithms, the network of the proposed algorithm is a stack of denoising auto-encoder, convolutional auto-encoder, and sparse auto-encoder with a more complex structure, so the average feature extraction time is slightly higher than the above two algorithms. As for the loop closure query time, since the proposed algorithm combines the advantages of different kinds of auto-encoders, it can extract features better and takes less time compared to SDA and CALC.

A loop closure detection algorithm based on SAAE is proposed to address the problem of limited precision and robustness of the traditional loop closure detection algorithm due to its poor generalization ability in different scenarios. The proposed SAAE is composed of multiple layers of denoising auto-encoder, convolutional auto-encoder and sparse auto-encoder. Using unsupervised

learning to train the model layer by layer, the well-trained model is used to learn the input images. It can complete low-dimensional to high-dimensional feature extraction and output features for loop closure detection. The test results on the new college and city centre datasets show that the algorithm in this paper is more robust than the traditional loop closure detection algorithm in different scenarios, which is still able to guarantee good precision at relatively high recall.

The task of loop closure detection faces a very complex dynamic environment. In future work, we will select more challenging datasets to train and test the proposed model, constantly adjusting the corresponding parameters to improve the performance of the algorithm. There is still a long way to go to fully apply the proposed SAAE into actual visual SLAM systems, and we wish to further investigate this idea in future research.

References

[1] DurrantWhyte Hugh F and Bailey Tim, Simultaneous Localization and Mapping. IEEE Robotics & Amp. Automation Magazine **13**, 99 (2006).

[2] Jorge Fuentes-Pacheco, JoséRuiz-Ascencio and Juan Manuel Rendón-Mancha, Artificial Intelligence Review **43**, 55 (2015).

[3] Labbe M and Michaud F, IEEE Transactions on Robotics **29**, 734 (2013).

[4] Shekhar R and Jawahar C V, Word Image Retrieval Using Bag of Visual Words, IEEE 10th IAPR International Workshop on Document Analysis Systems (DAS), 297 (2012).

[5] Cummins M and Newman P, International Journal of Robotics Research **30**, 1100 (2011).

[6] D. Galvez-López and J. D. Tardos, IEEE Transactions on Robotics **28**, 1188 (2012).

[7] E. Garcia-Fidalgo and A. Ortiz, IEEE Robotics and Automation Letters **3**, 3051 (2018).

[8] Liu Y and Zhang H, Visual Loop Closure Detection with a Compact Image Descriptor, IEEE/RSJ International Conference on Intelligent Robots and Systems, 1051 (2012).

[9] Zhang G, Lilly M J and Vela P A, Learning Binary Features Online from Motion Dynamics for Incremental Loop-Closure Detection and Place Recognition, IEEE International Conference on Robotics and Automation (ICRA), 765 (2016).

[10] Loukas Bampis, Angelos Amanatiadis and Antonios Gasteratos, The International Journal of Robotics Research **37**, 62 (2018).

[11] G. Zhang, X. Yan and Y. Ye, Loop Closure Detection Via Maximization of Mutual Information. IEEE Access, 124217 (2019).

[12] Liu Q and Duan F, Intelligent Service Robotics **12**, 303 (2019).

[13] Azam Rafique Memon, Hesheng Wang and Abid Hussain, Robotics and Autonomous Systems **126**, 103470 (2020).

[14] Gomez-Ojeda R, Lopez-Antequera M, Petkov N and Gonzalez-Jimenez J, Training a Convolutional Neural Network for Appearance-Invariant Place Recognition. Computer Science, 1505 (2015).

[15] Gao X and Zhang T, Autonomous Robots **41**, 1 (2017).

[16] Merrill N and Huang G Q, Lightweight Unsupervised Deep Loop Closure, arXiv:1805.07703, 2018.

[17] Burguera A and Bonin-Font F, Journal of Intelligent & Robotic Systems **100**, 1157 (2020).

[18] Fei Wang, Xiaogang Ruan and Jing Huang, IOP Conference Series: Materials Science and Engineering **563**, 052082 (2019).

[19] Aritra Mukherjee, Satyaki Chakraborty and Sanjoy Kumar Saha, Applied Soft Computing **80**, 650 (2019).

[20] Chen B, Yuan D and Liu C, Applied Sciences **9**, 1120 (2019).

[21] Gao Xiang and Zhang Tao, Fourteen Lectures on Visual SLAM: From Theory to Practice (2nd Edition), Beijing: Publishing House of Electronics Industry, 302 (2019).

[22] S. Lange and M. Riedmiller, Deep Auto-Encoder Neural Networks in Reinforcement Learning, The 2010 International Joint Conference on Neural Networks, 1 (2010).

[23] Vincent P, Larochelle H and Bengio Y, Extracting and Composing Robust Features with Denoising Autoencoders, Machine Learning, Proceedings of the Twenty-Fifth International Conference, 1096 (2008).

[24] Masci J, Meier U and Ciresan D, Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction. Artificial Neural Networks and Machine Learning (ICANN), International Conference on Artificial Neural Networks. 52 (2011).

[25] Zhang L, Lu Y and Wang B, Neural Process Lett **47**, 829 (2018).

[26] Jiang X, Zhang Y, Zhang W and Xiao X, A Novel Sparse Auto-Encoder for Deep Unsupervised Learning, Sixth International Conference on Advanced Computational Intelligence (ICACI), 256 (2013).

[27] Moacir Ponti, Josef Kittler, Mateus Riva, Teófilo de Campos and Cemre Zor, Pattern Recognition **61**, 470 (2017).

[28] Vincent P, Larochelle H and Lajoie I, Journal of Machine Learning Research **11**, 3371 (2010).

[29] Bordes A, Bottou, Léon and Gallinari P, Journal of Machine Learning Research **10**, 1737 (2009).

[30] Cummins M and Newman P, International Journal of Robotics Research **27**, 647 (2008).

[31] Kejriwal N, Kumar S and Shibata T, Robotics and Autonomous Systems **77**, 55 (2016).

[32] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva and A. Torralba, IEEE Transactions on Pattern Analysis and Machine Intelligence **40**, 1452 (2018).