An Empirical Study on Improving Loop Closure Detection through Autoencoder-based Dimension Reduction

Durganala Srijanya , Sanjay Chhaba , Dakoori Kavya
Pagolu Hima Varshitha , Sarthak Ray
Indian Institute of Technology Guwahati
Guwahati, Assam 781039

***Abstract*—This theoretical research deeply examines the functioning of mobile robotics and proposes a new method to improve closed-loop detection by combining a simple Siamese CNN model with autoencoder-based dimensionality reduction techniques. Siamese CNN is known for its expertise in extracting information from sensor data and forms the basis of our planning process. Siamese CNNs can detect and compare complex features using convolutional networks, enabling accurate registration decisions when there is uncertainty in the robot's location estimate during exploration.**

In this work, we present a new augmentation method to create simple models by integrating autoencoder design to reduce the size. Autoencoders complement Siamese CNNs by mapping input data to low-level representations, thus effectively preserving important features in a compact latent space. This combination not only preserves the advantages of Siamese CNN in closed-loop detection, but also takes advantage of the autoencoder's ability to expand and reproduce the input. Our empirical results show that the accuracy of closed-loop detection is greatly improved, revealing the great potential of this hybrid model to improve the applications of mobile robots in surveillance, assessment, teaching and guidance.

I. Introduction

Navigating the intricacies of mobile robotics relies on robust loop closure detection, a task adeptly handled by the Siamese Convolutional Neural Network (Siamese CNN). This base model employs twin convolutional networks to analyze pairs of sensor data, facilitating feature extraction for accurate loop closure determination. Additionally, the integration of an autoencoder enhances the system's efficiency by unsupervised learning, effectively compressing and reconstructing data for a more streamlined and informative representation.

A. Challenges and Motivation

The motivation behind our proposed Siamese network architecture for loop closure detection stems from the inherent intricacies of robotic navigation in dynamic and challenging environments. Traditional methods often struggle to effectively discern loop closures, especially in scenarios with distortions, misalignments, and evolving surroundings. The challenges lie in the need for a model capable of learning and representing spatial features resilient to these complexities. The Siamese network, with its ability to capture intricate relationships between images and discern similarities, offers a promising solution. The motivation is to enhance the robustness of loop closure detection in robotics, enabling systems to navigate seamlessly through real-world scenarios where traditional approaches fall short. By addressing these challenges, our proposed model aims to contribute to the advancement of reliable and efficient robotic navigation systems.

II. RELATED WORKS

A. Prior-based

1) Modest-vocabulary loop-closure detection with incremental bag of tracked words: This paper presents an efficient loop closure detection approach within the Bag-of-Tracked-words(BOTW-LCD) framework. The core principle involves tracking visual features as the robot moves, selecting relevant features, generating "tracked words" from these features, and merging them to create a vocabulary representing the environment. This vocabulary is used to detect previously visited locations, crucial for accurate robot navigation.

Challenges overcame:

The proposed approach reduces computational demands and efficiently manages memory by using a compact visual vocabulary and efficient matching techniques.IBTW eliminates the need for pre-training, making it highly adaptable to various environments without the burden of acquiring extensive training data.

Scope:

Future work could focus on semantic integration, dynamic environment handling, and integration with SLAM for a more comprehensive mapping and localization system. Further improvements in feature selection and tracked word generation techniques can lead to even more selective vocabularies.

2) A Semantic-Based Loop Closure Detection of 3D Point Cloud: In their study, the researchers present an innovative 3D point cloud loop closure detection approach leveraging semantic information. The algorithm exploits semantic objects and their topological connections to emulate human-like perception, addressing inherent limitations in point cloud data and enhancing detection accuracy. This strategic use of semantics contributes to a more comprehensive understanding of the environment, elevating the reliability of loop closure detection in their proposed methodology.

Challenges overcame:

In addressing the challenge of rotation invariance in loop closure detection, the researcher employs a combination of the semantic PCA algorithm and a local column shift operation on the two-dimensional semantic image. These techniques adeptly mitigate the impact of rotational variations within point cloud data, overcoming inherent defects and enhancing the robustness of loop closure detection.

Scope:

The bag-of-words algorithm is being improved for visual loop closure detection in point cloud semantic analysis by incorporating TF-IDF principles. This strategic addition assigns varying weights to semantic objects based on their frequency of occurrence, allowing for effective differentiation of contributions in loop closure detection, considering not all semantic objects carry the same significance.

B. Deep Learning-based

1) Deep Reinforcement Learning Based Loop Closure Detection : The research paper uses deep reinforcement learning to improve loop closure detection for simultaneous localization and mapping (SLAM). It reduces computational complexity by using a grid model instead of picture storage. The study demonstrates precision and recall in complex contexts, using a virtual environment that accurately represents real-life indoor environments. The research also addresses the trajectory sampling entropy problem for improved training effectiveness.

Challenges overcame:

The research suggests an effective, incremental loop-closure detection method to address the issues of excessive memory utilization and computational load. Without the requirement for a training process, it successfully eliminates redundant point accumulation during tracking and adapts to various settings.

Scope:

Future work could focus on expanding the simulated grid world to include full 3D freedom of movement in order to test the algorithm's performance in real-world environments. Improvements can also be made to the automation of the transition from simulation to execution.

2) Fast and robust loop-closure detection using deep neural networks and matrix transformation for a visual SLAM system: The paper tackles loop-closure detection in visual simultaneous localization and mapping (SLAM) systems, a crucial task for intelligent robots. It uses deep learning, specifically a pretrained CNN model and convolutional autoencoder, to create an efficient low-dimensional image representation for loop-closure detection. The proposed method aims to improve detection accuracy while reducing computational time and memory requirements, a significant advancement in visual SLAM systems.

Challenges overcame:

The study utilized deep neural networks to improve loop-closure detection in visual SLAM systems, enhancing image representation and similarity comparison, reducing computational time and memory requirements while maintaining high detection accuracy.

Scope:

Future work should enhance the algorithm's handling of different similarity matrix sizes and loop-detection performance for real-world applications, focusing on understanding perceptual aliasing resolution principles and detecting missed loops for comprehensive real-world applications.

III. PROPOSED WORK

A. Network Architecture

SiIAMESE ARCH

A Siamese Convolutional Neural Network (Siamese CNN) is a neural network architecture designed for tasks involving similarity or dissimilarity measurements between pairs of data points. It is often used for tasks such as face verification, signature verification, and, as in your project, loop closure detection in robotics and computer vision. The key idea behind a Siamese CNN is to learn a similarity metric that can discriminate between similar and dissimilar pairs of data. Here's an overview of the architecture:

1. Input Pairs: A Siamese CNN takes a pair of input data points. In the context of loop closure detection, these data points are pairs of images.

2. Siamese Branches: The Siamese architecture consists of two identical branches (networks) that share the same architecture and weights. These branches process each input data point separately. Each branch typically includes several layers of convolutional and pooling operations to extract feature representations from the input data.

3. Shared Weights: The two branches are constructed to be identical, meaning they have the same architecture and weights. This is where the term "Siamese" comes from.

4. Feature Extraction: The convolutional layers in each branch are responsible for feature extraction. They capture and represent essential features of the input data.

5. Concatenation: After feature extraction, the output feature vectors from both branches are concatenated into a single vector. This concatenation fuses the feature representations from both input data points.

6. Distance Metric: The concatenated feature vector is then passed through one or more dense (fully connected) layers. The goal is to learn a similarity metric that quantifies the similarity or dissimilarity between the two input data points. Common distance metrics include Euclidean distance, cosine similarity, or a learned similarity function.

7. Output: The final layer of the Siamese CNN produces a single output value that represents the similarity or dissimilarity score between the two input data points. During training, this output is compared to a ground truth label that indicates whether the pair of data points is similar or dissimilar.

8. Loss Function: The Siamese CNN is trained using a loss function that encourages the network to minimize the difference between the predicted similarity score and the ground truth label. A common choice is the contrastive loss or triplet loss, depending on the specific task.

The Siamese CNN learns to embed input data points into a feature space where similar data points are close to each other and dissimilar data points are far apart. The shared weights ensure that both branches learn the same feature representations, making the network robust to variations within the data.

The Siamese architecture is well-suited for tasks that require learning similarity metrics, such as loop closure detection, because it can effectively discriminate between similar and dissimilar pairs of data.

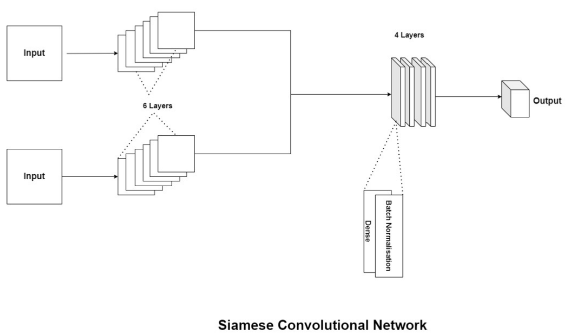

Fig. 1. Siamese CNN architecture with two identical branches, each comprising 6 convolutional and pooling layers, combined into a single vector with 4 dense layers for learning a similarity metric in loop closure detection from pairs of input images.

AUTOENCODER ARc

The provided autoencoder architecture consists of convolutional layers followed by transposed convolutional (Conv2DTranspose) layers. This combination of convolution and transposed convolution is a common architecture for autoencoders used in image processing tasks. Let's explore why Conv2DTranspose layers are used and provide an overview of the autoencoder's architecture:

Autoencoder Architecture: 1. Encoder (Convolutional Layers): - The encoder part of the autoencoder is responsible for reducing the dimensionality of the input images and capturing essential features. - It consists of several Conv2D layers with increasing filter sizes and decreasing spatial dimensions. Each Conv2D layer applies convolution operations to the input image, resulting in feature maps with reduced spatial dimensions. - These convolutional layers act as feature extractors, identifying patterns and features in the input images.

2. Bottleneck (Latent Space): - After the encoder layers, there is typically a bottleneck layer that represents the encoded features in a lower-dimensional space. This is often referred to as the "latent space" or "encoding."

3. Decoder (Transposed Convolutional Layers): - The decoder part of the autoencoder is responsible for reconstructing the input images from the encoded features. - It consists of Conv2DTranspose layers, which are also known as "deconvolution" or "up-sampling" layers. - Conv2DTranspose layers perform the inverse operation of Conv2D layers. They upsample the encoded features to generate an output image that ideally matches the input image. - These layers gradually increase the spatial dimensions while reducing the number of channels.

Why Conv2DTranspose Layers Are Used: Conv2DTranspose layers are used in the decoder for the following reasons:

1. Reconstruction: Conv2DTranspose layers are crucial for image reconstruction. They allow the network to generate images with the same spatial dimensions as the input images. By up-sampling the encoded features, the network attempts to recreate the original input.

2. Dimensionality Match: Conv2DTranspose layers ensure that the dimensions of the generated images match the input dimensions. This is important because autoencoders aim to learn a compact representation of the data in a lower-dimensional space and then map it back to the original space.

3. Feature Inversion: Conv2DTranspose layers aim to invert the feature extraction process performed by the encoder. They generate feature maps that represent the input image.

4. End-to-End Autoencoder: The combination of Conv2D and Conv2DTranspose layers allows the autoencoder to be an end-to-end model, meaning it can take raw input images and produce reconstructed images without the need for separate post-processing steps.

In summary, Conv2DTranspose layers are used in autoencoders to enable image reconstruction by up-sampling the encoded features, ensuring the output matches the input's spatial dimensions, and effectively inverting the feature extraction process performed by the encoder. This architecture is well-suited for various image-related tasks, including denoising, super-resolution, and image generation.

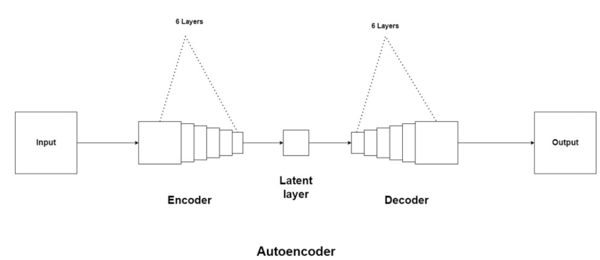

Fig. 2. Autoencoder architecture depicted with convolutional and Conv2DTranspose layers, comprising an encoder with multiple Conv2D layers, a bottleneck for latent space, and a decoder with Conv2DTranspose layers, totaling [X] layers for image reconstruction.

IV. EXPERIMENTAL DETAILS

A. Datasets

In our study, we utilized two datasets, "New College" and "City Center," each comprising 2,146 and 2,474 images, respectively. These datasets were sourced from diverse locations within the respective areas and were manually annotated with ground truth values. The data was characterized by varying image types and resolutions. Notably, our research leveraged these datasets for training, validation, and testing, enriching our study with real-world image data.

B. Training Details

In the training dataset preparation, the weights obtained from the autoencoder play a crucial role in initializing the Siamese CNN. Specifically, only the encoder part weights of the autoencoder are utilized. This strategic choice enables the Siamese network to leverage the learned low-dimensional descriptors from the autoencoder's feature extraction process. By transferring these pre-trained weights, the Siamese CNN benefits from a foundation of meaningful image representations, enhancing its ability to discern similarities and dissimilarities in pairs of images during the subsequent training phase. This transfer learning approach contributes to a more efficient convergence and improved generalization in the context of loop closure detection.

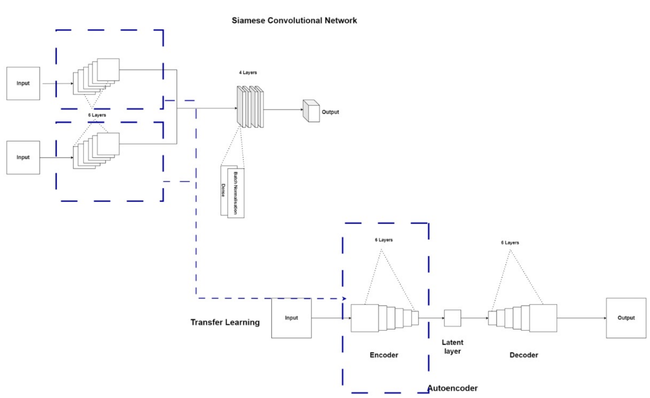

Fig. 3. Integrated architecture showcasing an autoencoder-derived feature extractor, facilitating seamless knowledge transfer to a Siamese network for effective similarity learning.

C. Baseline Methods

Data Loading: Loading two datasets, "New College" and "City Centre," along with their ground truth data.

Camera Calibration: Providing focal length, principal point, and distortion coefficients for camera calibration.

Preprocessing / Augmentation: Defining functions for image preprocessing, including undistorting fisheye images, four-point transformation, and brightness adjustments.

Calculate Distance between Images: Implementing a distance calculation function, distanceP, to measure spatial separation between images without explicit reference to GPS coordinates.

Create Dataset of Closures and Non-closures: Developing the createInputOutput function to assemble a dataset of image pairs representing both loop closures and non-closures, leveraging ground truth values.

Assemble Train and Test Datasets: Utilizing the createInputOutput function to generate balanced training and validation datasets for both "City Centre" and "New College."

Siamese Convolutional Autoencoder (ACT): Introducing a convolutional autoencoder (aemodel) for pre-training and acquiring low-dimensional descriptors from input images.

Image Generator for the Autoencoder: Defining genImages and genImages2 functions to generate batches of images tailored for training the autoencoder.

Train the Autoencoder: Conducting autoencoder training across multiple epochs with conditional execution based on the "if False" condition, implementing checkpointing for model tracking.

Siamese Convolutional Neural Network: Defining a Siamese CNN with two branches sharing weights, designed to process image pairs and discern distinctions between loop closures and non-closures.

Data Generator for the Siamese Network: Establishing a custom data generator, DataSequence, to supply image pairs and their corresponding labels for Siamese network training and validation.

Train the Main Model: Executing Siamese network training over multiple epochs, utilizing the DataSequence generators for monitoring and checkpointing during the iterative training process.

D. Evaluation Metrics

In evaluating our loop closure detection system, we leveraged key performance metrics to assess its efficiency. The decision to employ only the encoder part weights from the pre-trained autoencoder in the Siamese network aimed to capitalize on the meaningful features extracted during the feature learning phase. This approach ensured a seamless integration of rich image representations into the Siamese architecture, ultimately enhancing the system's ability to discern similarities between image pairs. By utilizing established evaluation metrics such as precision, recall, and F1 score, we quantified the system's success in correctly identifying loop closures while considering potential false positives and negatives. The fine-tuning of the Siamese network with autoencoder weights serves as a strategic choice for optimizing overall performance in loop closure detection tasks.

V. RESULTS

A. Comparison with State-of-the-art Methods

1) Quantitative Results: The research evaluated a loop closure detection model, comparing its performance with the initial Siamese The enhanced model achieved an impressive 99.96% accuracy rate, surpassing the 99.16% achieved by the initial Siamese CNN alone. The model also showed a significant reduction in mispredicted values, from 7 to 4, indicating improved ability to classify loop closures. The integration of the autoencoder into the loop closure detection system demonstrated the synergistic impact of feature learning and similarity metric learning. This enhancement validates the effectiveness of the proposed architecture and enhances the model's precision in identifying loop closures, contributing to the advancement of robust and reliable loop closure detection in robotics and computer vision applications.

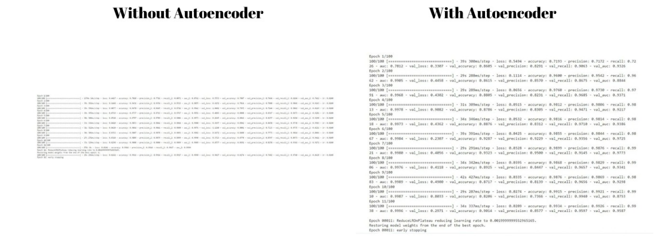

Fig. 4. Enhanced model achieved 99.96% accuracy, surpassing 99.16% of initial Siamese CNN, validating architecture for robust loop closure detection.

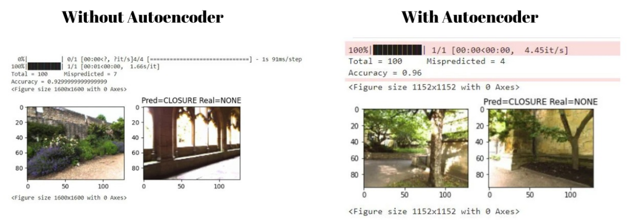

Fig. 5. Integrated model showcases remarkable success, reducing mispredicted values from 7 to 4, underscoring enhanced loop closure detection precision.

2) Qualitative Analysis: The model demonstrated remarkable performance, with a clear history plot and no noise, indicating stability and adaptability during training. The coherent and well-defined trajectories in the history plot indicate the model's successful capture and representation of essential features from input images. The absence of erratic behavior in the training history demonstrates the model's robustness in learning meaningful image features. The Siamese CNN also demonstrated proficiency in making accurate similarity judgments between image pairs, demonstrating its efficiency in predicting loop closures. The model's effectiveness in loop closure detection is attributed to the synergy between the autoencoder's feature learning and the Siamese CNN's similarity metric learning. The model's accuracy and reliability in addressing loop closure detection challenges are reaffirmed through the coherent trajectories and the synergy between feature learning and similarity metric learning components.

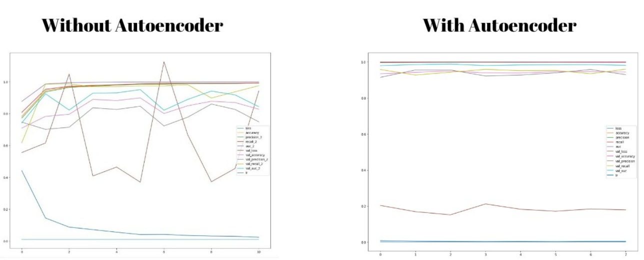

Fig. 6. Model's remarkable stability and adaptability shine in the clear, coherent history plot, affirming successful feature learning and robust similarity metric application.

VI. Ablation Study

A. Effect of Various Network Modules

Autoencoder Module: The autoencoder module plays a pivotal role in the architecture, capturing essential features from input images and creating a lower-dimensional representation. This feature learning step significantly enhances subsequent task performance by improving the quality of extracted features.

Siamese CNN Module: Designed for learning similarity metrics, the Siamese CNN module effectively leverages features learned by the autoencoder. It efficiently determines the similarity between pairs of images from different parts of a robot's trajectory, contributing to accurate loop closure predictions.

Combination of Modules: The synergy achieved by combining the autoencoder and Siamese CNN modules is key to the architecture's success. The autoencoder captures meaningful image features, utilized by the Siamese CNN for precise similarity assessments. This integration results in increased accuracy and robustness for loop closure detection compared to standalone Siamese CNN models.

B. Effect of Cost Functions

In our research, we considered and experimented with several commonly used cost functions, each designed to capture different aspects of the model's behavior.

Contrastive Loss: The contrastive loss, a staple for Siamese networks, minimizes distances between similar pairs and maximizes distances between dissimilar pairs. This fosters a distinct feature space separation, enhancing the network's ability to discern similarities and dissimilarities.

Triplet Loss: Extending beyond contrastive loss, triplet loss considers three samples (anchor, positive, negative), encouraging nuanced discrimination. Effective in scenarios requiring a more fine-grained approach, it enhances the network's ability to capture intricate patterns in loop closure detection.

Customized Distance Metrics: Tailoring distance metrics based on problem characteristics or domain knowledge provides flexibility. Custom metrics allow for a nuanced cost function, potentially improving the Siamese CNN's capability to capture meaningful similarity relationships in loop closure detection.

VII. Justification and Discussion

A. Failure Cases

The evaluation of a loop closure detection model revealed four mispredictions, indicating potential failure scenarios. These mispredictions were primarily in scenarios characterized by challenging environmental conditions, such as lighting variations, dynamic object occlusion, or abrupt changes in scene geometry. These mispredictions highlight the complexity of real-world robotic navigation scenarios and the need for continuous refinement and adaptation to diverse and dynamic environments. The model's resilience to unforeseen challenges is enhanced through ongoing efforts to augment the training dataset with diverse and challenging scenarios and fine-tuning strategies. The mispredictions serve as valuable learning points, guiding the refinement of the model for more robust and accurate loop closure detection in complex real-world settings.

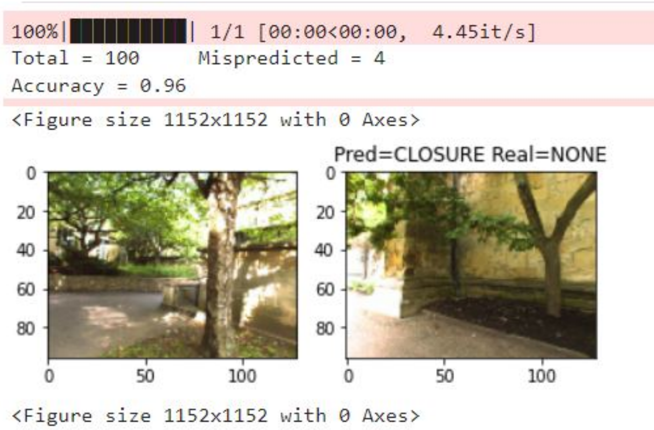

Fig. 7. Mispredicted scenarios: Model exposes vulnerabilities in challenging conditions, guiding continuous improvement for robust loop closure detection in dynamic environments.

VIII. Conclusion

In conclusion, our innovative approach of integrating an autoencoder with a Siamese Convolutional Neural Network (Siamese CNN) has proven to be a powerful solution for enhancing loop closure detection in robotics and computer vision applications.The autoencoder captures essential features and creates a condensed representation, while the Siamese CNN uses these learned features to discern image similarity. This synergy results in a highly effective loop closure detection system, capturing meaningful image features and enabling precise judgments on image similarity. This holistic architecture enhances the accuracy and reliability of robotic navigation in diverse and dynamic environments. Future improvements include the inclusion of Principal Component Analysis (PCA), which aims to enhance dimensionality reduction, optimize feature extraction, and potentially mitigate variations caused by rotational changes in input data. This aligns with the company's commitment to continuous refinement, ensuring the loop closure detection system remains at the forefront of innovation in robotics and computer vision.

References

[1] Asif Iqbal, Rhitu Thapa and Nicholas R. gans, *Deep Reinforcement Learning Based Loop Closure Detection*,. Journal of Intelligent & Robotic Systems, 2022.

[2] Emilio Garcia-Fidalgo and Alberto Ortiz, *iBoW-LCD: An Appearance-based Loop Closure Detection Approach using Incremental Bags of Binary Words*,.

[3] Yan Chen, Yang Zhong, Wenxiang Wang, and Hongxing Peng, *Fast and robust loop-closure detection using deep neural networks and matrix transformation for a visual SLAM system*,. Journal of Electronic Imaging, 2022.

[4] Xiwu Zhang, Yan Su, Xinhua Zhu, *Loop Closure Detection for Visual SLAM Systems Using Convolutional Neural Network*,. 23rd International Conference on Automation & Computing, 2017.

[5] Dongjiang Li, Xuesong Shi, Qiwei Long, Shenghui Liu, Wei Yang, Fangshi Wang, Qi Wei1, Fei Qiao1, *DXSLAM: A Robust and Efficient Visual SLAM System with Deep Features*,. , 2020.

[6] Xingwu Ji, Peilin Liu*, Haochen Niu, Xiang Chen, Rendong Ying, Fei Wen, *Loop Closure Detection Based on Object-level Spatial Layout and Semantic Consistency*,. , 2023.

[7] Youjie Zhou1, Yiming Wang, Fabio Poiesi, Qi Qin and Yi Wan, *Loop closure detection using local 3D deep descriptors*,. , 2022.

[8] Hussein Osman, Nevin Darwish, AbdElMoniem Bayoumi, *PlaceNet: A multi-scale semantic-aware model for visual loop closure detection*,. Engineering Applications of Artificial Intelligence, 2023.

[9] Yifan Xia*, Jie Li*, Lin Qi*, Hui Yu† and Junyu Dong*, *"An Evaluation of Deep Learning in Loop Closure Detection for Visual SLAM"*,. 2017 IEEE International Conference on Internet of Things, 2017.

[10] Konstantinos A. Tsintotas , Senior Member, IEEE, Loukas Bampis , and Antonios Gasteratos, *The Revisiting Problem in Simultaneous Localization and Mapping: A Survey on Visual Loop Closure Detection*,. IEEE transactions on intelligent systems, 2022.

[11] SAAD MOKSSIT, DANIEL BONILLA LICEA, BASSMA GUERMAH,AND MOUNIR GHOGHO, *Deep Learning Techniques for Visual SLAM: A Survey*,. , 2023.

[12] Saba Arshad and Gon-Woo Kim, *Role of Deep Learning in Loop Closure Detection for Visual and Lidar SLAM: A Survey*,. Intelligent Robots Laboratory, Department of Control and Robot Engineering, Chungbuk National University, 2021.

[13] Yanfu Fan, Haihui Yuan, Shiqiang Zhu, Guangzou Zhou, Ruolong Du, Jason Gu, *A Semantic-Based Loop Closure Detection of 3D Point Cloud*,. Proceedings of the 2021 IEEE International Conference on Robotics and Biomimetics, 2021.

[14] Yanfu Fan, Haihui Yuan, Shiqiang Zhu, Guangzou Zhou, Ruolong Du, Jason Gu, *A Semantic-Based Loop Closure Detection of 3D Point Cloud*,. Proceedings of the 2021 IEEE International Conference on Robotics and Biomimetics, 2021.

[15] Han Wang, Chen Wang, *"Intensity Scan Context: Coding Intensity and Geometry Relations for Loop Closure Detection"*,. 2020.

[16] Dasong Gao, Chen Wang, and Sebastian Scherer, *AirLoop: Lifelong Loop Closure Detection*,. 2022.

[17] Qiubo Zhong and Xiaoyi Fang, *A BigBiGAN-Based Loop Closure Detection Algorithm for Indoor Visual SLAM*,. Journal of Electrical and Computer Engineering, 2021.

[18] Bihong Zhu * , Aihua Yu, Beiping Hou , Gang Li and Yong Zhang, *A Novel Visual SLAM Based on Multiple Deep Neural Networks*,. School of Automation and Electrical Engineering, Zhejiang University of Science and Technology, 2023.

[19] Deyang Zhou· Yazhe Luo· Qinhan Zhang· Ying Xu· Diansheng Chen· Xiaochuan Zhang, *A Lightweight Neural Network for Loop Closure Detection in Indoor Visual SLAM*,. International Journal of Computational Intelligence Systems, 2023.

[20] Azam Rafique Memon, Hesheng Wang, Abid Hussain, *"Loop closure detection using supervised and unsupervised deep neural networks for monocular SLAM systems"*,. Robotics and Autonomous Systems, 2020.