Visual Loop Closure Detection with Thorough Temporal and Spatial Context Exploitation

Jiaxin Li*, Zan Wang*, Huijun Di, Jian Li† and Wei Liang†

***Abstract*— Despite advancements in visual Simultaneous Localization and Mapping (SLAM), prevailing visual Loop Closure Detection (LCD) methods primarily rely on computationally intensive image similarity comparisons, neglecting temporal-spatial context during long-term exploration. To address this issue, we propose *TOSA*, a novel visual LCD algorithm harnessing *TempOral and SpAtial context* for efficient LCD. Specifically, as the agent explores through time, our approach recurrently updates a latent feature incorporating historical information via a Long Short-Term Memory (LSTM) module. Upon receiving a query frame, *TOSA* seamlessly fuses the latent feature with the query feature to predict the candidates' distribution, thus averting intensive similarity computation. Additionally, *TOSA* integrates a *temporal-spatial convolution* for candidate refinement by thoroughly exploiting the temporal consistency and spatial correlation to enhance selected candidates, further boosting the performance. Extensive experiments across four standard datasets showcase the superiority of our method over existing state-of-the-art techniques, demonstrating the effectiveness of utilizing rich temporal-spatial contexts.**

I. Introduction

Loop Closure Detection (LCD), alleviating the accumulated dead-reckoning errors during long-term exploration, plays a crucial role in Simultaneous Localization and Mapping (SLAM) [8] systems. As the visual SLAM gains increasing attention due to its simplified sensor setup, low cost, and diverse applications in autonomous driving and robotics [27], visual LCD becomes increasingly pivotal within visual SLAM. Consequently, the pursuit of enhancing visual LCD performance through advanced image analysis technologies emerges as a prominent research avenue in computer vision and robotics communities [30, 32].

Existing methods [15, 19] detect the loop closures primarily in two steps: (i) *Candidate Proposal:* identifying candidate frames through one-by-one similarity comparisons; (ii) *Candidate Refinement:* applying the temporal consistency and geometrical verification to discern and retain the most reliable loop closure candidates. Despite their notable success and wide-ranging applications, these methods exhibit two fundamental limitations for further improving the computational efficiency and detection accuracy. First, prior works independently compare the image feature similarity between the query and historical frames to propose candidates, which thus neglects the rich temporal context information and necessitates intensive similarity computations, as depicted in Fig. 1 (a). Secondly, though these methods acknowledge the importance of *temporal consistency* by expecting adjacent queries to have adjacent loop closures, they overlook the fact that neighboring frames of a candidate are also likely to be candidates themselves, reflecting the *spatial correlation*.

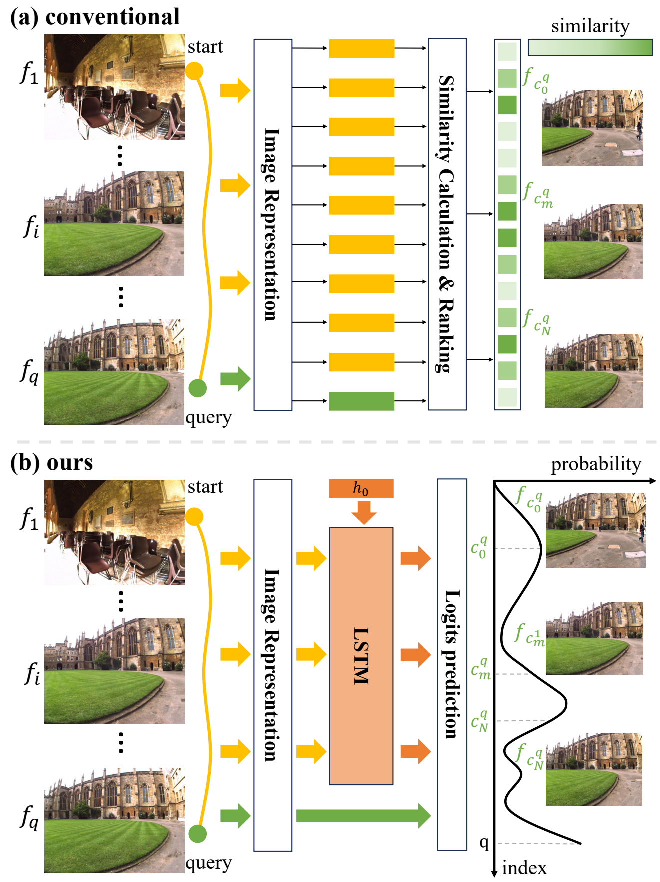


Fig. 1: **The comparison between our proposed method and conventional methods for proposing candidates.** **Yellow** curve represents the exploration trajectory. **Green** dot represents the current query frame. (a) Conventional methods compute the similarity between the query and historical frame pairs one by one. (b) Our method auto-regressively updates a latent feature, which memorizes the historical temporal context and contributes to the candidate distribution prediction.

To address the above issues, we introduce a novel framework for visual LCD, named ***TOSA***, which incorporates two distinct improvements in the two steps to detect the loop closure more efficiently and accurately. In the first step, we formulate the candidate identification as a multi-label classification task instead of relying on ranking frame similarity. Upon the formulation, we employ a LSTM to maintain a latent feature, which retains historical information

* Jiaxin Li and Zan Wang contributed equally to this work.
† Corresponding author.
All authors are from the Beijing Institute of Technology, Beijing, China
{lijiaxin, wangzan, ajon, lijian_551, liangwei}@bit.edu.cn.

and undergoes auto-regressive updating during long-term exploration. Once given a query frame, ***TOSA*** fuses the query and the latent features to directly predict the candidate distribution over historical frames, thus circumventing computationally expensive similarity comparisons, as illustrated in Fig. 1. The top-N frames with the probability exceeding a predefined threshold are selected as the candidates.

In the second step, we introduce a candidate refinement strategy that harnesses temporal consistency and spatial correlation to enhance the selected candidates. This strategy incorporates a novel *temporal-spatial convolution*, a convolution-like operation, to aggregate activations over the candidate proposal matrix from both temporal and spatial dimensions. We select the resulting candidate activations with scores surpassing a predefined threshold as the final loop closures. We extensively experiment across four standard datasets to demonstrate our method's consistent outperformance over existing SOTA techniques and conduct module ablations to showcase the effectiveness of each component.

Our contributions can be summarized as follows:

1) We propose to ***leverage the temporal-context information in an auto-regressive manner*** for efficient *Candidate Proposal*. We believe this new visual LCD paradigm could potentially inspire the community.
2) We introduce a novel ***temporal-spatial convolution*** operation to harness temporal consistency and spatial correlation, which improves the *Candidate Refinement*.
3) Through extensive experiments, we demonstrate the efficiency and effectiveness of our proposed method by surpassing the existing SOTA techniques.

II. Related Work

In numerous visual LCD methods, images are represented as global features. Oliva *et al*. introduce Gist [21, 22], which captures scene characteristics to represent images as features. Kazmi *et al*. [14] uses the weighted average of nearby locations' gist features for image representation. The histogram-of-oriented-gradients (HOG) [17] creates histograms based on pixel gradients. Unlike the aforementioned methods, Bag-of-Words (BoW) [19, 26] clusters local features to establish a "visual vocabulary" comprising quantified "visual words." When the incoming image enters, a visual word histogram is created based on the widely used TF-IDF [26] as the image feature. Meanwhile, VLAD [13] quantizes the differences between local features and neighboring visual words, concatenating the distances as image representation.

With deep learning developing fast, many methods utilizing Convolution Neural Network (CNN) for feature extraction have gradually emerged. Inspired by VLAD, NetVLAD [3] proposed a differentiable layer for image description via combing features extracted by base architecture such as VGG [25], ResNet [12], and MobileNetV2 [24]. Liu *et al*. [15] presents a strategy for self-supervised training of feature extractors utilizing motion knowledge, thus reducing labeling costs. VLASE [33] detects semantic edges for image description. FILD [1] uses a proximity graph structure for fast global feature searching, followed by SURF [5] feature extraction for geometrical verification. And FILD++ [2] utilizes a single network for global and local feature extraction.

The above image-to-image methods propose loop candidates by comparing the similarity between the query frame and each database entry. In contrast, sequence-based methods compute similarity based on sequential sub-maps, which consist of sequences of images [18] or image descriptors [28]. In SeqSLAM [18], likelihood scores are computed among the query sequence and database sequences at a predefined constant velocity. The path with minimum cost (*i.e*., summary of absolute differences) is regarded as the loop candidate. Vysotska *et al*. [31] utilizes graph optimization for such an alignment process, while Arroyo *et al*. [4] incorporates GPS priors for performance enhancement.

After identifying suitable candidates, geometrical verification is typically conducted using hand-crafted local features such as SIFT [16], SURF [5], ORB [23], or learned features [20] to filter the real loop closures.

However, these methods don't consider the intrinsic temporal context information within the traversed route and involve intensive similarity computation between image pairs or sequence pairs. In this work, we utilize the rich temporal context for the candidate proposal by fusing the query feature with a dynamic-updated latent feature. Additionally, we introduce a novel *temporal-spatial convolution* for candidate refinement, which thoroughly exploits temporal consistency and spatial correlation inherent in the exploration.

III. Method

Fig. 2 presents the framework of our proposed ***TOSA*** for LCD. In the *Candidate Proposal* stage, ***TOSA*** leverages the temporal context information in an auto-regressive manner to predict the distribution of loop closure candidates. Subsequently, ***TOSA*** incorporates a novel *temporal-spatial convolution* operation to thoroughly exploit the temporal-spatial information for *Candidate Refinement*.

A. Temporal Context Informed Candidate Proposal

Problem Formulation In LCD, the primary objective is to identify the most probable loop closures from a set of t historical frames, given a query frame f_q, where $q = t+1$. In this paper, we innovatively formulate the *Candidate Proposal* step as a multi-label classification task rather than solely relying on similarity comparisons, assigning N candidates to a query frame can be seen as predicting N labels.

Image Feature Extractor As depicted in Fig. 2, ***TOSA*** first extracts a global feature embedding e_i, serving as a global image descriptor, for each frame. Given the expectation that frames sharing similar appearances will yield comparable embeddings, we pre-train the CNN-based feature extractor using contrastive learning [11], which evaluates frame similarity via cosine similarity of the global features. However, directly training the feature extractor with loop closure annotations proves inefficient due to the sparsity of loop closure instances within a trajectory. Taking inspiration from Liu *et al*. [15], we extend our approach by considering neighboring frames of each frame as corresponding positive samples,

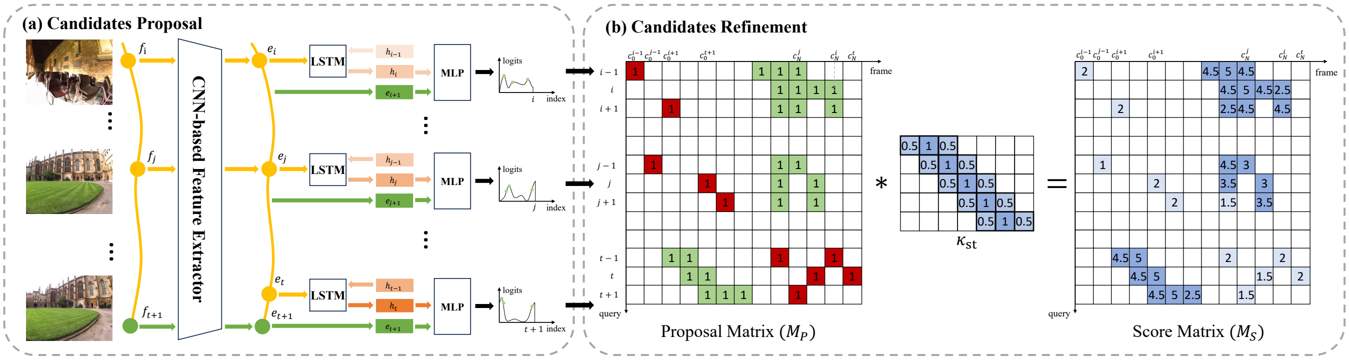


Fig. 2: **Overview of our proposed *TOSA*.** (a) We select N loop candidates, *i.e.*, $f_{c_0^{t+1}}, f_{c_1^{t+1}}, ..., f_{c_N^{t+1}}$, for the query frame f_{t+1} by leveraging temporal context. (b) We refine the loop candidates using a novel *temporal-spatial convolution* operation. Green proposals in the proposal matrix represent real loops. Red ones denote false positive proposals, which are filtered out by setting $\lambda_2 = 2$.

under the assumption that nearby frames are likely to exhibit the image similarity. The training loss is formulated as:

$$\mathcal{L}(e_i, e_j) = \mathbb{I}(i,j) \cdot \max(0, \epsilon - d)^2 + (1 - \mathbb{I}(i,j)) \cdot \max(0, \epsilon + d)^2,$$

where $\mathbb{I}(i,j)$ is an indicator function that outputs 1 when frame i and j are a pair. d represents the distance between the two extracted global features, computed using the cosine similarity of normalized features. ϵ acts as a decision threshold, enforcing a minimum separation between positive and negative pairs; we set $\epsilon = 0.8$ in our implementation.

Temporal Context Integration During the agent's long-term exploration, we maintain a latent state h_i to memorize the temporal context information through a LSTM module. Upon the arrival of the i-th frame, we directly forward the input frame embedding e_i and the last latent state h_{i-1} into the LSTM module, which yields an updated hidden state h_i, as shown in Fig. 2 (a). The initial latent state h_0 is a zero vector. When given a query frame f_{t+1}, we directly concatenate the query frame embedding e_{t+1} with the latent state h_t and forward the concatenation into a Multilayer Perceptron (MLP) for predicting the candidate distribution. To predict the multiple candidates for the query, we translate this multi-label classification problem into multiple binary classifications, *i.e.*, we predict a binary distribution for each historical frame. Label 1 denotes this frame is the candidate, and label 0 denotes this frame is not the candidate. This process is formulated as:

$$\mathbf{p}_0, \cdots, \mathbf{p}_t = \mathcal{G}(e_{t+1}, h_t),$$

where $\mathcal{G}$ represents the MLP network followed by a Sigmoid function; $\{\mathbf{p}_i\}_{i=1}^{t}$ are the binary distributions corresponding to historical frames.

During training, we optimize the parameters of both the LSTM and the MLP end-to-end without freezing the CNN-based feature extractor. The training loss is computed by averaging the Binary Cross-entropy over the historical frames, formulated as:

$$\mathcal{L} = \frac{1}{t} \sum_{i=1}^{t} \left(-\mathbb{I}(i) \cdot \log \mathbf{p}_i^1 - (1 - \mathbb{I}(i)) \cdot \log \mathbf{p}_i^0 \right),$$

where t represents the maximum number of classes, *i.e.*, the number of historical frames for a query to select as candidates. Notation $\mathbf{p}_i^1$ denotes the probability of frame f_i being the candidate of the query f_{t+1}, and vice versa for $\mathbf{p}_i^0$.

During inference, we rank the probabilities associated with the positive label for all frames. Ultimately, only K frames, with probabilities exceeding a predefined threshold λ_1, are selected as candidates from the top N frames.

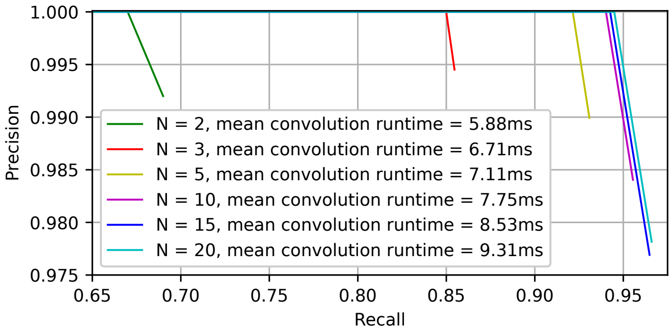
B. *Candidate Refinement with Temporal-Spatial Convolution*

Proposal Matrix After selecting candidates for the query frame f_{t+1}, we construct a candidate proposal matrix M_P using all historical query frames and their corresponding candidates. As depicted in Fig. 2 (b), M_P is a two-dimensional matrix where the horizontal dimension represents the frame dimension, and the vertical dimension represents the query dimension. Each entry $M_P(i,j)$ is a binary value, *i.e.*, 0 or 1. A value of 1 indicates that frame j is a loop closure candidate for query i, whereas a value of 0 means otherwise.

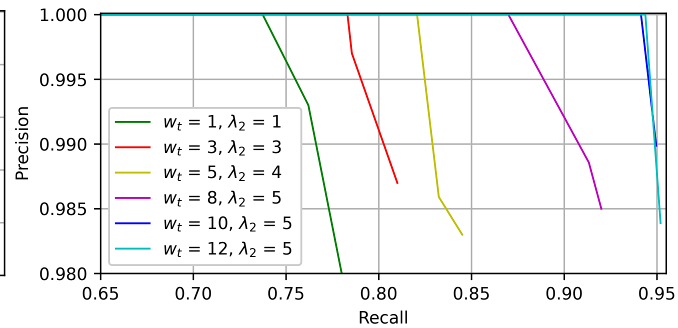
Temporal-Spatial Convolution and Score Matrix As illustrated in Fig. 2 (b), when a query frame f_t has a candidate frame $f_{c_1^t}$, the neighboring query frames $f_{t\pm1}$ are more likely to have candidate frames $f_{c_1^t \pm 1}$, indicating the *temporal consistency* along the query dimension. Similarly, in the frame dimension, if a frame $f_{c_1^t}$ is a candidate for a query frame f_t, then frames $f_{c_1^t \pm 1}$ are likely to be candidates for the same query frame, revealing the *spatial correlation*. To enhance the utilization of intrinsic temporal-spatial context information, we introduce a novel operation called *temporal-spatial convolution* to augment the selected candidates over the proposal matrix M_P. Concretely, *temporal-spatial convolution* utilizes a convolution kernel κ_{ts} to aggregate information from neighboring query frames (temporal dimension) and neighboring candidates (spatial dimension), producing a score matrix M_S. For the entry $M_P(i,j)$, the convolution operation with κ_{ts} is performed as follows:

$$M_S[i,j] = M_P\left[i - \frac{w_t}{2} : i + \frac{w_t}{2}, j - \frac{w_s}{2} : j + \frac{w_s}{2}\right] * \kappa_{st}.$$

Here, w_t and w_s represent the window size of the temporal and spatial dimensions, respectively, and $*$ denotes the convolution operation. Notably, due to the consistent temporal lag between candidates corresponding to separate



(a) Different N



(b) Different w_t

Fig. 3: **Precision and recall curves from analytical experiments on hyperparameters N and w_t.** Separately increasing both N and w_s results in improved performance. While, as N is increased to 10 and w_t to 10, the performance approaches saturation.

TABLE I: **Hyperparameters and corresponding default values.**

Hyperparameter	Value
Loop probability threshold, λ_1	0.9
Number of proposed candidates, N	10
Temporal window size of κ_{st}, w_t	10
Spatial window size of κ_{st}, w_s	3
Score threshold, λ_2	6

neighboring queries, we shift the kernel weight center of each row to reflect this characteristic.

After computing the score matrix M_S, we filter out candidates with scores below a predefined threshold λ_2. These retained candidates then undergo geometric verification for further confirmation. Candidates that pass the geometric verification are considered the final loop closures for the corresponding query frames.

C. Implementation

For implementation, we choose ResNet-50 [12] as the backbone for feature extraction, followed by two fully connected layers downgrading the output of ResNet-50 from 2048 to 16. LSTM module maintains a 512-dimension latent feature. To ensure the adaptability of our approach across various exploration distances, we maintain consistency by setting the output dimension of the MLP as $M = 4551$ for all evaluation datasets. We employ zero-padding for datasets with shorter sequence lengths to ensure compatibility. We summarize the selected important parameters utilized in our experiments in Tab. I. Default values are employed during the experiments unless explicitly specified.

IV. Experiment

In this section, we introduce four publicly available datasets used for evaluation. Next, we perform a series of analytical experiments and present comparative results against state-of-the-art methods, showcasing our method's effectiveness. Finally, we analyze our method's execution time and memory usage, crucial for its practical deployment.

A. Datasets

To evaluate our proposed method, we conducted extensive experiments on four publicly available datasets, including NewCollege (NC) [6], CityCentre (CC) [6], and two sequences from the KITTI dataset [10], namely, K00 and K05. NC and CC contain 1073 and 1273 pairs of images captured by two cameras arranged alternatively, respectively. CC is specially designed to assess the ability to match images in the presence of scene changes. On the other hand, K00 and K05 contain 4541 and 2761 images, respectively, captured by a monocular camera. Additional details about these datasets are provided in Tab. II. We use the ground truth annotations from Cummins *et al.* [6] for NC and CC datasets, where the loop closures are manually labeled. As for K00 and K05, the annotations are provided by Zhang *et al.* [34]. Our feature extractor was first pre-trained on the Places365 dataset [35] and fine-tuned using contrastive learning on the corresponding evaluation set. We use 50% frames of each dataset for fine-tuning and overall end-to-end training.

TABLE II: **Statistics of the datasets used for evaluation.**

Datasets	Description	Images	Size
NC [6]	Outdoor, Dynamic	2146, 0.5Hz	640 × 480
CC [6]	Outdoor, Dynamic	2474, 0.5Hz	640 × 480
K00 [10]	Outdoor, Dynamic	4551, 10Hz	1241 × 376
K05 [10]	Outdoor, Dynamic	2761, 10Hz	1226 × 370

TABLE III: **Recall at 100% precision of different w_s.**

w_s	2	3	4	5
Recall	92.61%	94.14%	93.55%	93.79%

B. Methods Analysis

In this section, we conducted the following analytical experiments on NC datasets.

We first assess the impact of varying N, which ranges from 2 to 20, on the final loop closure detection. As depicted by the precision and recall curves in Fig. 3a, there is a notable performance improvement as N increases from 2 to 10. However, beyond this threshold, the rate of performance improvement diminishes, indicating that $N = 10$ is sufficient for detecting loops. Moreover, it's important to note that the average time required for the subsequent convolution operation increases proportionally with N. Therefore, we opt to utilize $N = 10$ for the following experiments.

Next, we assess the impact of the window size of

TABLE IV: **Ablation studies on contrastive learning and candidate refinement.** Our full model achieves the best result.

Model	Recall	Precision
w/o contrastive learning	90.50%	100%
w/o candidate refinement	98.12%	84.29%
TOSA (full)	94.14%	100%

TABLE V: **Recall at** 100% **precision of different models.** The **bold** number indicates the best result, and the underlined number represents the second-best result. * denotes that the number of images used in the New College dataset by Tsintotas *et al.* [29] and FILD [1] differs from those used in other methods.

Methods	NC*	CC	K00	K05
FABMAP 2.0 [7]	52.63	40.11	61.22	48.51
SeqSLAM 2.0 [18]	66.67	75.12	78.33	61.48
Tsintotas *et al.* [29]	16.30	52.44	93.18	<u>94.20</u>
DLoopDetector [9]	47.56	30.59	72.43	51.97
FILD [1]	76.74	66.48	91.23	85.15
Liu *et al.* [15]	<u>91.21</u>	86.01	93.02	92.53
Ours (MobileNetV2)	90.39	<u>87.52</u>	<u>94.62</u>	93.68
Ours (ResNet-50)	**94.14**	**90.82**	**95.12**	**96.29**

temporal-spatial convolution kernel. In Tab. III, we present the recall values corresponding to different w_s, with w_t set to 10 and adjusting λ_2 to achieve 100% precision. The model utilizing $w_s = 3$ outperforms the one with $w_s = 2$ by a considerable margin, underscoring the beneficial impact of incorporating spatial information when $w_s > 2$. However, models using $w_s = 4$ and $w_s = 5$ showcase an inferior performance compared to $w_s = 3$. This is attributed to false positive candidates within the w_s window. While optimizing λ_2 helps exclude these false positives, it inadvertently excludes some real loops. Based on these findings, we select $w_s = 3$ for the subsequent experiments.

Fig. 3b presents the precision and recall curves for different values of w_t, with w_s set to 3 and each row of the kernel containing values $0.5, 1, 0.5$. We adjust λ_2 for optimal performance for each model. The results show that selecting $w_t = 10$ yields comparable results to $w_t = 12$, suggesting that $w_s = 10$ adequately captures temporal information. The green curve corresponds to $w_t = 1$, indicating that candidates are refined based solely on spatial information. This model achieves 73.73% recall at 100% precision, significantly lower than the 94.14% recall achieved by the model with $w_t = 10$, implying that the negative proposals attain high scores based on spatial information alone. Additionally, we observe that the optimal λ_2 for models employing $w_t = 8$, $w_t = 10$, and $w_t = 12$ is 5, suggesting that the upper bound of λ_2 for this type of false positives is 5 when $w_s = 3$. Thus, $w_t = 8$ is sufficient for selecting true positives while excluding false positives. The performance improvement achieved by increasing w_t to 10 further underscores the untapped potential of temporal information. We select $w_t = 10$ as the default.

To evaluate the effectiveness of key modules in our design, including the *constrastive learning* and *candidate refinement*, we ablate these components separately and compare them against our full model. Experimental results are presented in Tab. IV. Compared to the model without contrastive learning, denoted as "w/o contrastive learning," our full model ***TOSA*** achieves a higher recall rate at 100% precision, indicating the efficacy of optimizing the global feature extractor through contrastive learning. Notably, the model without candidate refinement achieves a high accuracy of 98.12%, demonstrating that the temporal information captured by the LSTM across the entire historical sequence significantly aids in identifying actual loop closures for each query frame. However, the precision is unsatisfactory: 993 query frames are classified as having loop closures, compared to the actual 853. Our proposed candidate refinement strategy significantly enhances precision while minimally missing a few true positives. This trade-off highlights the strategy's effectiveness in improving precision while maintaining high recall rates.

TABLE VI: **The mean execution time on NC dataset.**

Steps	Mean Time
Global Feature Extraction	34.6ms
Latent Feature Update	0.5ms
Loop Distribution Calculation	0.7ms
Top-N Selection	0.9ms
Candidate Refinement	8.0ms
Geometrical Verification	25.1ms
Whole System	69.8ms

C. Comparative Result

This section presents a comparative analysis of our method against several well-known state-of-the-art techniques, including FABMAP 2.0 [7], SeqSLAM 2.0 [18], Tsintotas *et al.* [29], DLoopDetector [9], An *et al.* [1], and Liu *et al.* [15]. Additionally, we introduce a variant implemented with MobileNetV2 [24] for image feature extraction. Comparison results regarding recall at 100% precision are presented in Tab. V. The hyperparameters, except for λ_2 adjusted to 8.5 specifically for evaluations on the CC dataset, remain consistent across all evaluation datasets, as outlined in Tab. I. The results demonstrate the superior performance of our proposed ***TOSA*** across four datasets compared to these state-of-the-art methods. Moreover, the variant of ***TOSA*** utilizing MobileNetV2 consistently outperforms these techniques on CC and K00 datasets while exhibiting a comparable performance on NC and K05. This outcome suggests that our method maintains its efficacy with a lighter, potentially less powerful, but more efficient feature extractor.

D. Execution Time and Memory Usage Analysis

We analyze the execution time and memory usage by implementing our method with Python on an Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz machine, along with an NVIDIA Geforce RTX 3090 GPU. We utilize the variant based on MobileNetV2 for time consumption evaluation. We summarize the execution time of each specific step in VI. The analysis reveals that the majority of the computational time is attributed to global feature extraction and geometrical verification. The average execution time of the entire system aligns with the requirements for real-time operation, achieving a performance over 14Hz. For memory usage, ***TOSA*** implemented with ResNet-50 occupies around 122.88 MB, while the MobileNetV2 variant occupies around 34.18 MB. On average, ***TOSA*** with ResNet-50 consumes up to 3 GB of memory during the inference across multiple tests.

V. Conclusion

We propose a new LCD method, *i.e*. ***TOSA***, which thoroughly exploits the intrinsic temporal and spatial context information. Dividing the method into a *Candidate Proposal* and a *Candidate Refinement* stage, we innovatively formulate the first stage as a multi-label classification task. To address it, we propose to leverage the temporal information across the entire historical sequence through a LSTM module to avert intensive similarity computations and facilitate detection accuracy. In the second stage, we introduce a novel *temporal-spatial convolution*, which further harnesses the temporal consistency and spatial correlation, effectively improving the precision while maintaining the recall. Our extensive experiments across four datasets demonstrate the superiority of ***TOSA*** over existing state-of-the-art techniques.

Limitations and Future Works Given the computational expense involved in global feature extraction and the tendency of LSTM to overlook early historical information, our forthcoming efforts will enhance the feature extraction network's efficiency and augment our method's long-range modeling capability.

References

[1] S. An, G. Che, F. Zhou, X. Liu, X. Ma, and Y. Chen, "Fast and incremental loop closure detection using proximity graphs," in *International Conference on Intelligent Robots and Systems (IROS)*, 2019.

[2] S. An, H. Zhu, D. Wei, K. A. Tsintotas, and A. Gasteratos, "Fast and incremental loop closure detection with deep features and proximity graphs," *Journal of Field Robotics*, vol. 39, no. 4, pp. 473–493, 2022.

[3] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[4] R. Arroyo, P. F. Alcantarilla, L. M. Bergasa, and E. Romera, "Towards life-long visual localization using an efficient matching of binary sequences from images," in *International Conference on Robotics and Automation (ICRA)*, 2015.

[5] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European Conference on Computer Vision (ECCV)*, 2006.

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

[7] M. Cummins and P. Newman, "Appearance-only slam at large scale with fab-map 2.0," *International Journal of Robotics Research (IJRR)*, vol. 30, no. 9, pp. 1100–1123, 2011.

[8] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part i," *IEEE Robotics & Automation Magazine (RAM)*, vol. 13, no. 2, pp. 99–110, 2006.

[9] D. Gálvez-López and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *Transactions on Robotics (T-RO)*, no. 5, pp. 1188–1197, 2012.

[10] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[11] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[13] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[14] S. A. M. Kazmi and B. Mertsching, "Detecting the expectancy of a place using nearby context for appearance-based mapping," *Transactions on Robotics (T-RO)*, vol. 35, no. 6, pp. 1352–1366, 2019.

[15] B. Liu, F. Tang, Y. Fu, Y. Yang, and Y. Wu, "A flexible and efficient loop closure detection based on motion knowledge," in *International Conference on Robotics and Automation (ICRA)*, 2021.

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

[17] C. McManus, B. Upcroft, and P. Newman, "Scene signatures: Localised and point-less features for localisation," *Robotics: Science and Systems X*, pp. 1–9, 2014.

[18] M. J. Milford and G. F. Wyeth, "Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights," in *International Conference on Robotics and Automation (ICRA)*, 2012.

[19] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *Transactions on Robotics (T-RO)*, vol. 33, no. 5, pp. 1255–1262, 2017.

[20] H. Noh, A. Araujo, J. Sim, T. Weyand, and B. Han, "Large-scale image retrieval with attentive deep local features," in *International Conference on Computer Vision (ICCV)*, 2017.

[21] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision (IJCV)*, vol. 42, pp. 145–175, 2001.

[22] A. Oliva and A. Torralba, "Building the gist of a scene: The role of global image features in recognition," *Progress in Brain Research*, vol. 155, pp. 23–36, 2006.

[23] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *International Conference on Computer Vision (ICCV)*, 2011.

[24] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[25] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[26] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *International Conference on Computer Vision (ICCV)*, 2003.

[27] F. Tang, H. Li, and Y. Wu, "Fmd stereo slam: Fusing mvg and direct formulation towards accurate and fast stereo slam," in *International Conference on Robotics and Automation (ICRA)*, 2019.

[28] K. A. Tsintotas, L. Bampis, S. An, G. F. Fragulis, S. G. Mouroutsos, and A. Gasteratos, "Sequence-based mapping for probabilistic visual loop-closure detection," in *IEEE International Conference on Imaging Systems and Techniques (IST)*, 2021.

[29] K. A. Tsintotas, L. Bampis, and A. Gasteratos, "Assigning visual words to places for loop closure detection," in *International Conference on Robotics and Automation (ICRA)*, 2018.

[30] K. A. Tsintotas, L. Bampis, and A. Gasteratos, "The revisiting problem in simultaneous localization and mapping: A survey on visual loop closure detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 19 929–19 953, 2022.

[31] O. Vysotska and C. Stachniss, "Effective visual place recognition using multi-sequence maps," *RAL*, vol. 4, no. 2, pp. 1730–1736, 2019.

[32] Y. Wu, F. Tang, and H. Li, "Image-based camera localization: an overview," *Visual Computing for Industry, Biomedicine, and Art*, vol. 1, no. 1, pp. 1–13, 2018.

[33] X. Yu, S. Chaturvedi, C. Feng, Y. Taguchi, T.-Y. Lee, C. Fernandes, and S. Ramalingam, "Vlase: Vehicle localization by aggregating semantic edges," in *International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[34] X. Zhang, L. Wang, Y. Zhao, and Y. Su, "Graph-based place recognition in image sequences with cnn features," *International Journal of Robotics Research (IJRR)*, vol. 95, no. 2, pp. 389–403, 2019.

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