Received July 22, 2019, accepted July 31, 2019, date of publication August 12, 2019, date of current version August 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2934521

A Novel Loop Closure Detection Method Using Line Features

RUIFANG DONG[1], ZHAN-GUO WEI[2], CHANG-AN LIU[3], AND JIANGMING KAN[1]
[1]Key Lab of State Forestry Administration for Forestry Equipment and Automation, School of Technology, Beijing Forestry University, Beijing 10083, China
[2]School of Transportation and Logistics, Central South University of Forestry and Technology, Changsha 410004, China
[3]School of Control and Computer Engineering, North China Electric Power University, Beijing 102206, China

Corresponding author: Jiangming Kan (kanjm@bjfu.edu.cn)

This work was supported by the Fundamental Research Funds for the Central Universities of China under Grant BLX201819.

ABSTRACT Loop closure detection is a significant requirement for simultaneous localization and mapping (SLAM) to recognize revisited place. This paper presents a novel line-based loop closure detection method for vision-based SLAM that allows reliable loop closure detections, especial under structural environment. The performance of coping with perceptual aliasing conditions is more competitive than point based methods. The bag of words model is extended in this work which uses only line features. A variant of TF-IDF (term frequency & inverse document frequency) scoring scheme is proposed by adding a discrimination coefficient to improve the discrimination of image similarity scores, further to reinforce the similarity evaluation of two images. LBD (Line Band Descriptor) and binary LBD features are extracted to build visual vocabularies. Temporal consistency and spatial continuity checks enhance detection reliability. The performance of proposed scoring scheme was compared with original TF-IDF, results show that our proposed scheme has competitive discrimination ability. We also compared the query performance of our vocabularies with ORB-based, MSLD (mean standard-deviation line descriptor)-based, and PL (Point-and-Line)-based vocabularies, results indicate that our vocabularies obtain the highest successful retrieval rate. The performance of the whole loop closure detection algorithm was also evaluated in terms of precision, recall and efficiency, which were compared with ORB, MSLD, PL-based methods, and also with CNN-based method, results demonstrate that our method is superior to others with satisfactory precision and efficiency.

INDEX TERMS Vision-based SLAM, bag of words, binary LBD, LBD, a variant of TF-IDF scoring scheme.

I. INTRODUCTION

Simultaneous localization and mapping (SLAM) has received a lot of attention in the robotic community during past years. It is well known as the problem of synchronously estimating the map of the environment and getting localized in it by exteroceptive sensors [1]. In particular, vision-based SLAM uses camera as exteroceptive sensor. One of the significant requirements for SLAM is loop closure detection, which means the ability of a robot to recognize places it visited before [2]. Correct loop closure detections will provide correct data association, which is beneficial to obtain a more accurate and consistent map. Specifically, for vision-based SLAM techniques, which are based on pose graphs, the detected loops can insert additional constraints into the graph, to improve estimation results [3], [4]. Despite significant progress in visual loop closure detection, challenges remain especially in illumination, viewpoint changing conditions and dynamic environments.

This work focuses on loop closure detection for vision-based SLAM, especially for the visual SLAM under structural environment, such as indoors, urbans, outdoors with buildings, transmission lines, etc. In the loop closure detection literature, there are three main types of methods have been proposed: ‘‘map-to-map’’, ‘‘image-to-map’’ and ‘‘image-to-image’’. ‘‘Map-to-map’’ approaches found corresponding features by considering their relative 3D location in world, they assumed that the past motion estimation was precise enough to find a loop [5], [6]. While ‘‘image-to-map’’ type try to associate features between the latest collected image and a retained spatial representation of the already-visited scene. The ‘‘image-to-image’’ methods (or appearance-based methods) detect correspondences use the observation data directly. They compare the similarity

The associate editor coordinating the review of this article and approving it for publication was Zhenhua Guo.

between the current image and past key-frames. A loop can then be detected in that of high similarity [7].

The most common type is appearance-based loop closure detection methods [8]–[10]. The basic idea is to construct a database of images captured during the travel, so as to retrieve the most similar image when a new one is collected [11]. The bag of words (BOW) model has been widely used in appearance-based methods [12]–[15]. This model builds a ''dictionary'' beforehand in an offline process through clustering visual feature descriptors extracted from a large number of training images. When a feature (e.g. SIFT, SURF) is extracted from the image, its descriptor is approximated by the entries in the dictionary. If the feature appeared in the dictionary, it is claimed to be a visual ''word''. By this means, we can use a vector of numerical visual words to represent an image, that is visual-word-vector. Thereby, the similarity of two images can be measured through calculating the difference of two related vectors. These methods result in very effective and quick solutions [16].

However, they have some limitations. Firstly, perceptual aliasing and perceptual variability will mislead mapping and localization seriously around very similar scenes, which is always occurs around structural environments, such as urbans and indoors, etc. Secondly, the main features used are point features such as SIFT or SURF, they are low level features that are far from describing complex structures [17].

Due to the inherent dimensionless character that point features are far from representing the environmental structure. Some work used lines as the features of SLAM, e.g. [18]–[24]. Line features are abundant in much man-made structural environment, such as indoors, transmission towers, buildings, etc. They can convey structural information effectively, because a 3D line spans over a higher level space than a 3D point as [25] described. Furthermore, line segments matching can be accomplished even when viewpoints have big changes. In spite of the advantages of lines, however, they have not been widely used as points. It owns to the higher difficulty of tracking lines than points. In addition, for lack of reliable descriptors that it is difficult to use in loop closure detection. To our best knowledge, few of loop closure detection work uses line features. Until now, only [25] used line only features, they were described by MSLD (the mean standard-deviation line descriptor) [26]. Different from [25], this paper applies LBD (line band descriptor) [27] to describe line feature. It has been verified that LBD outperforms the MSLD in terms of efficiency, accuracy and robustness. In order to improve the efficiency of loop closure detection, this work also converts LBD into a binary form, named as binary LBD descriptor.

On the other hand, based on BOW model and consider the drawback of TF-IDF, this work modifies TF-IDF scoring scheme by adding a discrimination coefficient to improve the discrimination of visual word. The proposed scoring scheme is named TDI(term frequency & discrimination coefficient & inverse document frequency) scheme. Finally, temporal consistency and spatial continuity checks similar to [11] are used to solve the perceptual aliasing and variability problems.

To summarize, the main contributions of this paper include:

(1). Line features based visual vocabularies are built with binary LBD and LBD descriptors. The query performance was compared with ORB, MSLD, and PL based vocabularies.

(2). A variant of TF-IDF scoring scheme named TDI is proposed to improve the discrimination of visual words, thereby to enhance the accuracy of similarity evaluation for two observations.

(3). A whole loop closure detection algorithm that applies only line features is proposed, especially work for man-made environment. Temporal consistency and spatial continuity checks enable reliable loop detection. Experiments were carried out to validate the properties of this algorithm.

The remainder of this paper is organized as follows. Section II introduces related work. Section III illustrates the detection and description of line features. Section IV presents the vocabulary constructing and our scoring scheme. In Section V, we introduce the loop determination. Section VI gives experiments and analysis. Section VII describes conclusions and future work.

II. RELATED WORK

This section discusses some of the most representative approaches in the field of appearance-based loop closure detection, in terms of how it is related to our method reported. Readers can investigate extended survey through the work of [28].

A. VISUAL REPRESENTATION OF SCENE

In computer vision and loop closure detection literatures, scenes observed are usually represented by visual features. The visual features can be divided into two types: global and local features [29].

Global features are extracted from entire image, they can encode original image pixels, shape and color data. Such as GIST descriptors, they are constructed from responses of Gabor filters from different orientations and scales, it was used in [30], [31] for loop closure detection. The SeqSLAM [32], SeqSLAM2.0 [33] used the sum of absolute differences between contrast low-resolution images as global features to perform sequence-based place recognition. It showed better performance under severe environment changes. However, global features have not flexibility, and they are more susceptible to change in the viewpoint and occlusion than local features.

In contrast, local features express local information of a patch centered at each interest point, line, patch, etc. Most work applied point features as local features, such as SIFT, SURF, ORB, etc. Due to an image can contain hundreds of local point features, the BoW model is often used as a quantization technique for them in order to construct a feature vector of an image. For example, this model used SIFT features to detect loops in [34], FAB-MAP [35] applied it for the SURF features, RTAB-Map SLAM [36] utilized it for both SIFT and SURF features, [11] used BOW to for ORB features, it showed promising performance of loop closure detection. However, they perform poorly when illumination conditions change. As mentioned previous, line features have competitive advantages compared with point features in structural environment, and in the conditions with occlusion or changeable viewpoint and. However, line features have not been widely used.

Different from above mentioned methods that use only one kind of feature modality, some methods [23], [29] combined two or more features to represent scene. For example, [23] combined point and line features as local features. However, when various features are combined, some of them may be redundant or suboptimal, that a careful weighting scheme is necessary to integrate different features.

Recently, the advantages of deep convolutional neural networks (CNNs) prompted loop closure detection community to explore them as a potential solution to cope with the weaknesses of hand-crafted features. Reference [37] used ConvNet features as global features of image, which showed better discrimination ability than hand-crafted global features, such as GIST, etc. References [38]–[41] used the output of particular CNN layers as descriptors to operate loop closure detection. Although CNN-based methods show better retrieval performances, they are still decoupled from the loop closure and SLAM functionalities. References [42], [43] told that the CNN's rely on viewpoint invariant appearances, and the shortage of topological information at the higher network levels make them as suboptimal for loop closure detection work. In contrast, local features based approaches are widely applied in visual SLAM, they can be easily fused with an illumination invariant image representation method to thereby prompt their robustness to possible environment changing.

B. DETERMINATION OF LOOP

Given an input image and the scene representations of previous visited places, the most similar ones need to find to further recognize the revisits, and determine the loop.

Some of loop closure detection methods are based on image matching, they look for the most similar individual image as the loop position. Such as some CNN-based methods [3], [37], [41], they chose the most similar one as the loop. However, these methods typically suffer from the perceptual aliasing problem when the robot is in the similar scene or indoors. In addition, as the robot visits more and more places, storage requirements will increase and search speed will decrease. If the BOW model is employed, image retrieval can be enhanced by using inverted indices, such as FAB-MAP2.0 [7], and the method proposed in [11].

To enhance the precision of loop detection, [35] presented a probabilistic framework to estimate the possibility that two images been collected at a same place. This framework is effective. A generative model of appearance is trained in an offline process, approximating the possibilities of co-occurrences of the words included in visual dictionary. Followed this scheme, [34] proposed a solution in a Bayesian filtering framework. In which color histograms are merged into the dictionary as visual features. It means that two visual vocabularies (point features and color histograms) are combined as input of Bayesian filter to compute the matching probability between two images. The solution considers the matching probability of past observations. Different from the probabilistic methods, [11] proposed a temporal consistency check technique to take into account past matches. Also, [11] chose to support their detection by gathering similarity scores from a lot frames collected close together in time. In general, succeeding images are regarded as sequences of multiple visual-word-vectors, these sets of visual-word-vectors are compared with the database and given an extra score. The high reliability of this loop closure detection has been proved in [44], [45]. Recently, [14], [33], [46] used a sequence of images to characterize a place in order to improve the robustness under significant environmental changes due to variations in weather, daylight and season.

III. LINE FEATURE DETECTION AND DESCRIPTION

This work uses Line Segment Detector (LSD) [47] to detect line segments from observed images. LSD is an epidemic line detection method, it benefits from its linear-time, accurate results and without requirement of parameter tuning. In this work, the length of line segments is selective, lines shorter than 20 pixels are unacceptable.

This work describes line feature based on LBD [27]. LBD is robust to image transformation due to its multi-scale line detection algorithm. With the designed descriptor, LBD is fast to calculate. Also, it can obtain high efficiency. Own to the pairwise geometric consistency assessment that it is precise even for low-texture images. Given a line segment in image, the line direction d_L and the orthogonal direction $d_\perp$ can be determined. These two directions in further to construct a local 2D coordinate frame. This local coordinate frame assigns the middle point of line segment as origin. The direction that is clockwise orthogonal to d_L is determined as orthogonal direction $d_\perp$. A local rectangular region centered at the line and aligned with the directions d_L and $d_\perp$ is selected as line support region (LSR). The descriptor is calculated from the LSR. Set the length of line to be l, the total width to be W, split this LSR into m bands, the direction of each band is parallel to the line. Each band is said to be a sub-region of LSR, and with length l and width $w = W/m$. [27] evaluated the performance of different values of m and w, when $m = 9$ and $w = 7$, it achieved the best performance. Thus, this work adopts this pair of parameters. For each band, build a band description matrix (BDM) by accumulating the gradients of pixels in each row and arranging the four collected gradients of all rows in stacks. In further, each band descriptor (BD) can be established by calculating the mean vector and the standard deviation vector of BDM as an 8-dimensional vector. Finally line band descriptor LBD is simply formed by concatenating the m band descriptors (BDs). Therefore, LBD is able to describe different-length

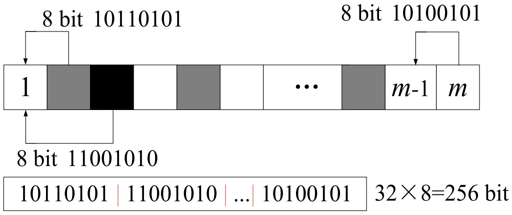


FIGURE 1. **Binary LBD descriptor.**

lines with uniform $8 \times m$ dimensional vectors. In this work $m = 9$, LBD is a 72-dimensional vector.

In order to improve calculation efficiency of subsequent procedures, we convert the LBD into a binary form, which consists of a series of 0 or 1, these 0 or 1 encodes the comparison results of this 72-dimensional vector. It is said to be binary LBD descriptor. It is converted as Figure 1 shows. Due to LBD vector consists of m ($m = 9$ in this work) BD vectors, each BD vector is an 8-dimensional vector, we extract 32 pairs from the m BD vectors according to a particular order. For example, the 1st BD vector and 2nd BD vector make a pair. For each pair of 8-dimensional BD vectors, we compare the data of each dimension, then we can get an 8-bit string. Then a 256-bit final binary LBD vector can be obtained by concatenating 32 comparison 8-bit strings.

Both 256-bit binary LBD descriptor and 72-dimensional LBD descriptor can be employed directly to build visual vocabulary. We compared the performance of these two different ways in experimental section.

IV. IMAGE DATABASE

Image database is always applied in loop closure detection to store and retrieve images to compare similarity. In [11], it consists of a hierarchical bag of words, direct and inverse indexes.

A. VISUAL VOCABULARY OF LBD

Bag of words is able to convert an image into a sparse numeric vector by using visual vocabulary. It is beneficial to manage big set of images. A large number of visual features are used to train the visual vocabulary offline, by dividing the feature space into W visual words. Reference [16] constructed the vocabulary as a hierarchical tree. It significantly improved retrieval quality and efficiency, as well as enabled the use of a larger vocabulary.

To build the vocabulary tree, features extracted are divided into k groups at first. Then, k-means clustering with k-means ++ seeding is executed for each group to obtain its center. In further, each group is composed of the features closest to cluster center. In vocabulary tree, the cluster centers work as the first level of nodes. Next, perform the same operation recursively for each group of features, and each group is also be split into k new sub-groups, in further to generate nodes recursively. By this way, the vocabulary tree is established level by level until to a predefined L times. Finally a vocabulary tree with $N = (k^{L+1} - k)\big/(k-1)$ nodes, and $W = k^L$ leaf nodes is established. The leaf nodes form the words of vocabulary. Finally, a weight should be assigned to each visual word to compute the relativity of a database image to a query image.

This work extracted a rich set of binary LBD descriptors from a great quantity of training images to build our vocabulary tree. Contrast to LBD descriptors, a binary feature space will obtain a more compressed visual vocabulary.

TF-IDF (term frequency & inverse document frequency) is a prevailing approach to determine the weight of each word. It is based on the relevance of words in training set. The words with high ‘‘term frequency’’ have higher weights, but the words with ‘‘high inverse document frequency’’, which are very frequent and less discriminative, are penalized by reducing weights. Although the well-known TF-IDF has been proved to be an effective scheme for term weighting in information retrieval, it is not the most effective one to reflect the importance of word. This work slightly modified original TF-IDF scheme to enhance discrimination of words to make retrieval more accurate.

We define that query image is converted into vector $\boldsymbol{q}$ by the bag of words model, and database image is converted into vector $\boldsymbol{d}$, then the weight of word i can receive the score q_i and d_i with original TF-IDF as (1),(2):

$$q_i = n_i w_i \tag{1}$$

$$d_i = m_i w_i \tag{2}$$

where, n_i and m_i are the number of word i in the query and database image, respectively. w_i indicates inverse document frequency, computed by (3), N represents the total number of training images, N_i is the number of training images which includes word i.

$$w_i = \ln \frac{N}{N_i} \tag{3}$$

Based on the idea of TF-IDF, we consider the case that when a visual word has high ‘‘term frequency’’ in both database and query images, it also has close ‘‘term frequency’’ in both images. In this case, this word is able to receive a high weight and make a great contribution to the similarity between database and query image. However, to some extent, this word has low discrimination in aspect of the number, but it receives a high weight. Consider this case, we propose a variant of TF-IDF scheme, TDI (term frequency & discrimination coefficient & inverse document frequency) method, by adopting a discrimination coefficient (DC) into the weight calculation equation of word as (4),

$$q_i = n_i d_{ci} w_i \tag{4}$$

where d_{ci} is the DC of word i.

The DC is calculated based on coefficient of variation (CV). As is well-known, the coefficient of variation (CV) is a statistical measure of dispersion of a data series around the mean. It can always be applied to compare the variability of two or more data series. A large CV value means that the data series is more variable, less steady or less uniform.

While a small CV means that the data series is less variable, much steadier or more uniform. CV is commonly used in the evaluation field to compute the weight of index. The index will receive a high weight when the related CV value is high. Similar to this usage, we add DC to compute the weight of visual word in order to enhance the similarity discrimination between query image and database images.

Assume we have database image1, image2, ..., image Y. Consider a simple and extreme situation, only word 1, 2, 3 are the same words extracted in all of images, the number of word 1, 2, 3 are shown in Table.1. For word 1, the number in each image is 10, different from word 1, word 2 has a larger span in word number, and word 3 has the largest span. It means although word 1 has a high term frequency, however, it has low discrimination. In contrast, word 3 has a lower term frequency, but with a high discrimination. Therefore, in the aspect of discrimination, word 3 will receive a higher weight and word 1 will receive a lower one. Following this idea, the CV of word number is adopted to compute the weight of visual word. However, it is not suitable to use CV directly. The reason is following.

TABLE 1. The number of words in database images.

Database image / Word ID	Database image 1	Database image 2	Database image 3	...	Database image Y
1	10	10	10	...	10
2	6	8	7	...	9
3	2	6	7	...	8

Our analysis starts from the definition of CV as (5) shows, where c_v represents CV, σ and μ is the standard deviation and the mean of data series. In aforementioned case, for word 1, the standard deviation of word number is 0, it results in $c_v = 0$, this result is not appropriate for our usage. Therefore, we propose discrimination coefficient (DC).

$$c_v = \frac{\sigma}{\mu} \tag{5}$$

DC is computed as follows.

Suppose there are Y training images, the number of visual words is X, the number of times that word x appeared in image y is $b_{xy}, y \in \{1, 2, \cdots, Y\}$. Then, the CV of each word can be computed as $c_{vx}, x \in \{1, 2, \cdots, X\}$. It is noted that when we compute c_{vx} for word x, the image with $b_{xy} = 0$ is not taken into account. Finally, DC is obtained by (6), (7). In our case, σ_x is the standard deviation of b_{xy}, and μ_x is the mean of b_{xy}.

$$d_{cx} = \begin{cases} \xi_0, & c_{vx} = 0 \\ \xi_0 + \varepsilon \dfrac{c_{vx}}{c_{v\min}}, & c_{vx} \neq 0 \end{cases} \tag{6}$$

$$X\xi_0 + \varepsilon \sum_{x=1}^{X} \frac{c_{vx}}{c_{v\min}} = 1 \tag{7}$$

where $c_{v\min} = \min\{c_{v1}, c_{v2}, \cdots, c_{vX}\}$, ε is a coefficient defined by user, then according to (7), ξ_0 can be achieved.

It must be noted that query and database bag of words vectors must be normalized.

B. SCORING

We use L_1-norm to compute the similarity of two image bag of words vectors, the similarity value lies in [0-1]. Inverse index proposed in [11] is maintained to speed up the retrieval process. Inverse index stores the id-numbers of images in which a specific word appeared, as well as the TDI weight for each image. This way reduces the comparisons against database images, due to the comparison process is limited to the images that have a few same words with the query image. And the inverse index will be updated when a new image is inserted to the database.

V. LOOP CLOSURE DETECTION

The loop closure detection includes two steps, feature processing and loop determination. Feature processing involves line features extracting and descriptors calculating. Loop determination consists of three operations: (1) transforming an image into a bag of words vector; (2) calculating similarity scores between database images and query image; (3) the find of candidates and determination of loop. Previous sections have introduced feature processing, the operation of converting an image into a bag of words vector and calculating similarity scores between database images and query image. This section will describe the selection of candidate loops and the determination of loop.

A. VISUAL VOCABULARY OF LBD

Define the image captured at time t as $\mathbf{I}_t$, $\mathbf{I}_t$ can be transformed into a bag of words vector v_t. The similarity score between image $\mathbf{I}_{t1}$ and $\mathbf{I}_{t2}$ equals to the similarity score between v_{t1} and v_{t2}, this similarity score is represented by $s(v_{t1}, v_{t2})$. When an image is obtained as query image $\mathbf{I}_t$, the retrieval process is performed among database images $\mathbf{I}_k$. Then a lot of corresponding similarity scores $s(v_t, v_k)$ are acquired. A threshold γ_t is necessary to determine the acceptable candidate similarity score. It is variable depending on the query image and the words appeared. In this work, γ_t is defined by (8),

$$\begin{cases} \gamma_t = \alpha \cdot s \\ s = s(v_t, v_{t-1}) & \text{if } s(v_t, v_{t-1}) \geq s(v_t, v_{t-2}) \\ s = s(v_t, v_{t-2}) & \text{if } s(v_t, v_{t-1}) < s(v_t, v_{t-2}) \end{cases} \tag{8}$$

where, α is a coefficient used to multiply the best similarity score to adjust the threshold. In general, the previous image has the best similarity score with $\mathbf{I}_t$. However, the case when occlusion and blur occur can bring a small similarity score. Thus, instead of $s(v_t, v_{t-1})$, we use a higher similarity score of $\mathbf{I}_t$ with previous two images.

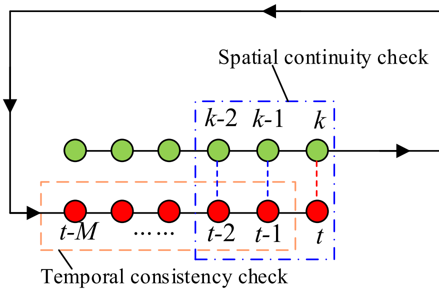


FIGURE 2. Temporal consistency check and spatial continuity check.

Here, the image which has a similarity score more than γ_t is selected as a candidate, otherwise, it is rejected.

In addition, due to the images that are adjacent always have high similarities, this work performs temporal consistency check. It means the previous M images $\mathbf{I}_{t-M} \sim \mathbf{I}_{t-1}$ are not taken into count to be loop candidates. As Figure 2 shows, red circles $t - M \sim t$ represent the image frames $\mathbf{I}_{t-M} \sim \mathbf{I}_t$, $\mathbf{I}_{t-M} \sim \mathbf{I}_{t-1}$ which lie in orange dotted box are not considered as loop candidates.

B. DETERMINATION OF LOOP

The following two steps similar to [11] are used to determine the loop closure from candidate frames.

In general, adjacent images always have high similarity, it may cause the competition among the close loop candidates. To avoid this condition, candidates are grouped based on the consecutive image ID as an island. And each island gets a match score by accumulating the similarity scores of each candidate within the group. The island that obtains the highest similarity score is chosen as loop group. The image gets the highest score in loop group will be selected as the best candidate loop frame, it will pass into next step.

To enhance reliability of loop detection, a spatial continuity check is designed as the final step by making comparison with previous queries. The best candidate loop frame must be consistent with previous loop frame. It means the best candidate loop frame should be close to a few previous loop frames. As shown in Figure 2, green circles k, k-1, k-2 represent the frames $\mathbf{I}_k$, $\mathbf{I}_{k-1}$, $\mathbf{I}_{k-2}$, $\mathbf{I}_t$ is current frame, $\mathbf{I}_k$ is the best candidate loop frame. $\mathbf{I}_t$ can be determined as loop frame when $\mathbf{I}_{k-2}$ and $\mathbf{I}_{t-2}$, $\mathbf{I}_{k-1}$ and $\mathbf{I}_{t-1}$(connected by the two blue dotted lines) are determined as the corresponding loops. If the best candidate loop frame pass this check, it will be accepted as the final loop frame.

VI. EXPERIMENTAL RESULTS AND ANALYSIS

This section describes experimental results to evaluate the performance of presented method in different aspects. Our datasets are divided into training group and test group, as shown in Table.2 and Table.3. Training group includes 4 public datasets, which covers a variety of indoor and outdoor conditions. Test group includes 4 public datasets and 3 sequences captured in NCEPU (North China Electric Power University) by ourselves. They also covers a variety of indoor and outdoor, static and dynamic conditions. All of datasets cover man-made environment. All of experiments used same settings. The training group was employed to train different vocabularies (i.e. ORB, MSLD, LBD, binary LBD). Each vocabulary was trained with $k_w = 10$ branches and $l_w = 5$ depth levels that is 100 thousand words.

TABLE 2. Training datasets.

Datasets	Description	Camera position	Image size (px×px)
Malaga6L[48]	Outdoors, dynamic	Frontal	1024×768
QUT Gardens Point campus [49]	Indoors, static	Frontal	1280×960
New College[50]	Outdoors, dynamic	Lateral	640×480
Euroc[51]	Indoors, static	Frontal	752×480

TABLE 3. Test datasets.

Datasets	Description	Camera position	Image size (px×px)
NCEPU, Main TB, C part	Morning, indoors, static	Frontal	640×480
	Night, indoors, dynamic	Frontal	640×480
NCEPU, No. 2 TB	Indoors, dynamic	Frontal	640×480
Kitti[52]	Outdoors, dynamic	Frontal	1241×376
TUM[53]	indoors, static	Frontal	640×480
Hanyang University [54]	Outdoors, indoors, dynamic	Frontal	640×480
City Centre	Outdoors, dynamic	Lateral	640×480

The first experiment compared the discrimination of proposed TDI and original TF-IDF weighting scheme. The second experiment compared the query performance of vocabularies trained using ORB, MSLD, LBD, binary LBD and PL(Point-and-Line) features (point descriptor is ORB, line is described by binary LBD), we used the scheme of [23] to compute the similarity score based on PL features. Finally, we carried out loop closure detection experiments to validate our proposed algorithm.

A. THE DISCRIMINATION COMPARISON

This paper proposes a TDI weighting scheme to improve the discrimination of visual words, thereby to improve the calculation accuracy of image similarity. In this experiment, we captured five images with 1280×960 pixels as Figure 3 shows. There were four books on the desk in Figure 3(a). We removed the books one by one as Figure 3(b)-(e) show, the reason we took images by this way is that it is intuitive to evaluate similarity. Take Figure 3(a) as reference image, we computed the similarity scores between Figure 3(a) and Figure 3(a)-(e).

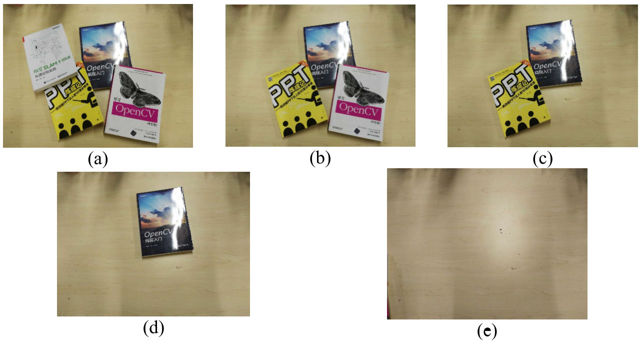

(a) (b) (c)
(d) (e)

FIGURE 3. The images for comparing discrimination. (a) shows the reference image, in (b) - (e), the number of books reduce in turn.

TABLE 4. The comparison of discrimination.

Similarity score	Descriptors			
	ORB		LBD	
	TF-IDF	TDI	TF-IDF	TDI
$s(v_a,v_a)$	1	1	1	1
$s(v_a,v_b)$	0.0473826	0.0738264	0.122766	0.3156145
$s(v_a,v_c)$	0.0464083	0.0568406	0.0780863	0.2282616
$s(v_a,v_d)$	0.0202128	0.0228142	0.0226661	0.1058402
$s(v_a,v_e)$	0.00319021	0.00292342	0.00117477	0.00222521
Similarity score	Descriptors			
	Binary LBD		MSLD	
	TF-IDF	TDI	TF-IDF	TDI
$s(v_a,v_a)$	1	1	1	1
$s(v_a,v_b)$	0.272472	0.4053416	0.330246	0.4243531
$s(v_a,v_c)$	0.148678	0.2152157	0.312558	0.2954211
$s(v_a,v_d)$	0.0434117	0.0985632	0.248346	0.1043545
$s(v_a,v_e)$	0	0.00067224	0.00681811	0.00032145

The similarity scores computed using TF-IDF and proposed TDI for 4 different descriptors are shown in Table 4. We can see that for each descriptor, the highest similarity scores of both TDI and TF-IDF schemes are received between reference image and itself, which reaches 1. The similarity score decreases in turn from (b) to (e). This result is consistent with our intuitive sense, when the number of books decreases, the similarity score reduces too. But in terms of discrimination, Table.4 shows that TDI has a more obvious gap for all of 4 descriptors. Take MSLD descriptor as an example, for TF-IDF, the differences between $s(v_a, v_b)$ and $s(v_a, v_c)$, $s(v_a, v_c)$ and $s(v_a, v_d)$, $s(v_a, v_d)$ and $s(v_a, v_e)$ are 0.017688, 0.064212, 0.24152789. While for our TDI, they are 0.128932, 0.1910666, 0.10403305, respectively. This result indicates that TDI weighting scheme obtains a higher discrimination than TF-IDF. It is beneficial to evaluate similarity.

B. EVALUATION OF VOCABULARY

The construction of visual vocabulary is a crucial work for loop closure detection. This section evaluates the performance of constructed visual vocabularies using proposed method. For comparison, the performance of vocabularies trained using ORB, MSLD, LBD, binary LBD and PL features were evaluated. Both TF-IDF and TDI methods were compared. Similar to [25], the performance was evaluated applying stereo images and monocular images which gathered twice under different illuminations. Among stereo images, left images were regarded as database images, right images worked as query images. It is assumed that there is an exact alignment between left and right images. For monocular images, two sequences were gathered by a same camera in 8th floor of NCEPU, main TB, C Part. But under different illumination situations (in morning and at night) and with dynamic people. During capturing, robot kept moving with a constant velocity. It traveled a loop, then at the end of loop, robot continued to move about 10 m. Meanwhile, at night, the robot followed the path it passed in morning. The database was established using image sequence collected at night (368 images), and the morning image sequence (362 images) worked as queries.

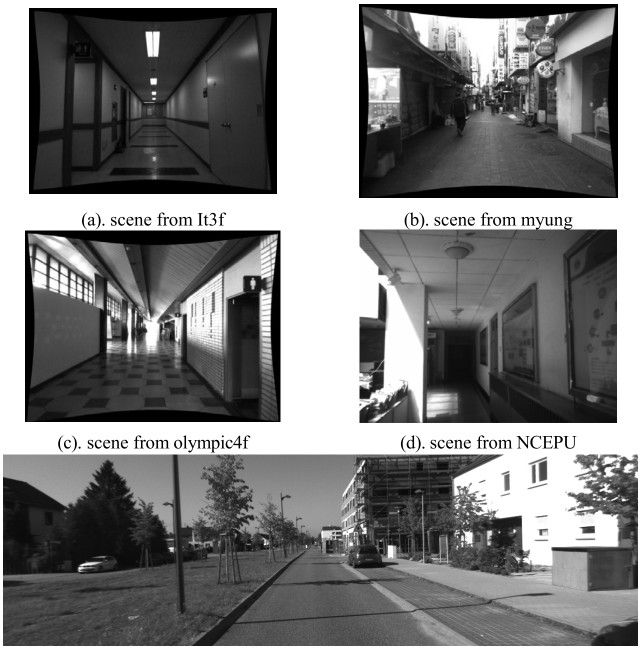
We computed successful retrieval rate to evaluate the performance of proposed method. The successful retrieval rate is defined by the number of successful query frames against the total number of query frames. A successful retrieval is defined as: when at least one of the top 3 results queried is not farther than two frames from ground truth.

The stereo images used include It3f, myung, olympic4f, kitti06 and the sequence gathered by ourself in 1st floor of NCEPU, No. 2 TB (we will use "TB2" to describe it for brevity). It3f, Myung and olympic4f sequences come from public dataset Hanyang University. As Fig.4 shows, the stereo datasets applied include outdoor and indoor scenes, there are a lot of structured features, such as buildings, trees, shops, roads, cars, doors, windows, ceilings, etc.

Table.5 shows retrieving results for stereo images using TF-IDF and TDI scheme for different features, i.e. ORB, MSLD, LBD, binary LBD and PL.

TABLE 5. The query results.

TF-IDF					
Features	Successful retrieve rate				
	it3f	myung	olympic4f	kitti06	TB2
ORB	84.53%	76.07%	58.64%	86.00%	80.84%
binary LBD	86.70%	88.28%	91.40%	98.62%	84.23%
LBD	90.40%	87.74%	95.32%	100.00%	85.12%
MSLD	87.05%	89.19%	91.78%	99.12%	84.25%
PL	85.16%	88.55%	87.78%	98.50%	84.18 %
TDI					
Features	Successful retrieve rate				
	it3f	myung	olympic4f	kitti06	TB2
ORB	88.23%	85.22%	78.43%	92.62%	90.54%
Binary LBD	93.22%	95.98%	95.26%	99.75%	93.63%
LBD	95.69%	95.61%	97.27%	100.00%	94.62%
MSLD	94.57%	96.22%	95.47%	99.62%	93.76%
PL	91.24%	93.97%	92.79%	97.56%	93.57%



(a). scene from It3f

(b). scene from myung

(c). scene from olympic4f

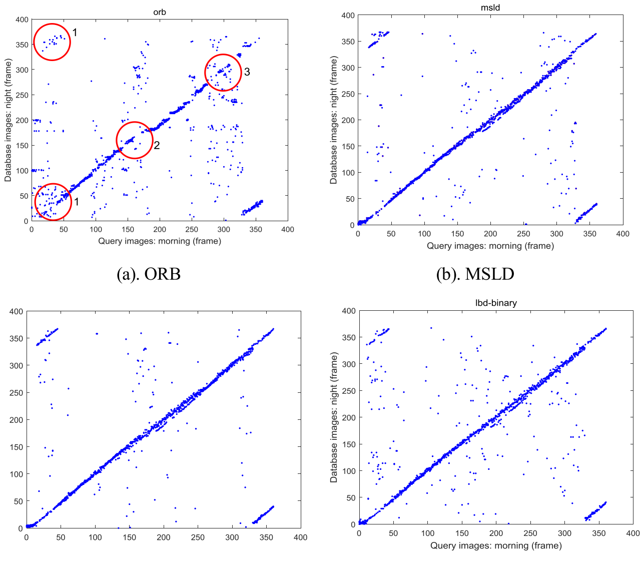
(d). scene from NCEPU

(e). scene from kitti06

FIGURE 4. The scenes in It3f, myung, olympic4f, kitti06 and NCEPU sequences.

TABLE 6. Query results for various conditions.

Method	Successful retrieve rate				
	ORB	Binary LBD	LBD	MSLD	PL
TF-IDF	44.72%	65.72%	69.05%	66.61%	45.93%
TDI	60.45%	84.44%	88.15%	84.46%	61.56%



(a). ORB

(b). MSLD

(c). LBD

(d). binary LBD

FIGURE 5. The query results under different illuminations (night and morning) and dynamic scenes with (a) ORB, (b) MSLD, (c) LBD, and (d) binary LBD descriptors.

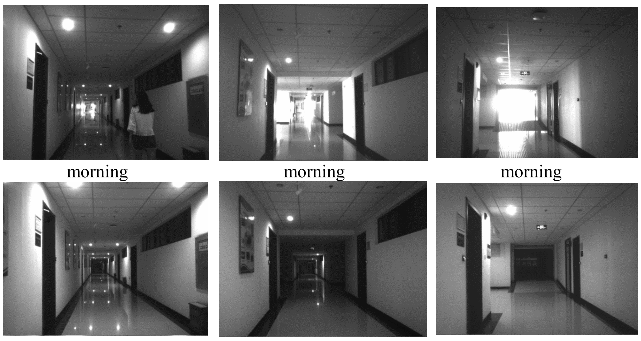
From Table.5, we can see that for all features, the proposed TDI is superior to TF-IDF scheme in terms of successful retrieval rate. And all of line descriptors are superior to point descriptors. PL feature reveals a better performance than point feature, but it is a bit inferior to line features, it is because in the calculation of similarity score, both point and line features affect the results, the weights of them must be set carefully. Among line descriptors, LBD achieved highest successful retrieval rate, binary LBD descriptor has a related lower successful retrieval rate, but it is very close to LBD and MSLD's results.

Table.6 shows the retrieving results for two monocular sequences captured as described beforehand using both TF-IDF and TDI schemes.

Table.6 shows that illumination changes and dynamic people affect retrieval results. Line descriptors are superior to point descriptors. But the successful retrieval rates of queries used TDI scheme are still acceptable. Figure 5 plots the retrieval results by different vocabularies using TDI. x-axis stands for the ID of query images, y-axis represents the ID of database images. We just plot query results within top 5 similarity scores. From the plots, we can see a diagonal, linear distribution of dots. And on top-left and bottom-right of the plots, there are two symmetrically short lines, they are images in query database which related to the revisited places in establishing database. Affected by the illumination changes and dynamic people, the distributions of points in Figure 5 along the diagonal, top left and bottom right are dispersive. But in Figure 5(a), the plotted points highlighted within red circles are more dispersive than Figure 5(b)-(d), it means that the queries used ORB descriptors failed in these places. Figure 6 shows image pairs captured in these places (position 1c $\sim$ 3). We can see that retrieval results in positions 2 and 3 are affected by natural light, the result in position 1 is affected by dynamic people. As shown in Figure 6, the scene is structured, with a lot low-texture regions, results indicate that it is better to employ line features than points in such scene.

C. EVALUATION OF LOOP CLOSURE DETECTION

The proposed loop closure detection algorithm was evaluated using Precision-Recall curves. "Precision" is defined as the ration between the number of correctly detected loop closure frames (True-Positive) and the total number of detected loop closure frames (True-Positive and False-Positive). "Recall" is defined as the ratio between the number of correctly detected loop closure frames (True-Positive) and the total number of true loop closure frames (True-Positive and False-Negative) that exist in sequences. Precision-Recall curve of loop closure detection was obtained by changing normalized similarity threshold α which introduced in section IV.A, α was set from 0.05 to 0.99. Note that the ground truth of loop closures were computed through the corresponding odometry data. The proposed algorithm were compared with ORB,



morning morning morning

night night night

(a). position 1 (b). position 2 (c). position 3

FIGURE 6. **The scenes to test vocabulary performance under scenes with illumination changes and dynamic people. The image pairs in (a) position 1, (b) position 2, and (c) position 3 are scenes acquired from the corresponding positions on the Figure 5(a) marked with red circles. And the images captured in the morning are on the top, the images captured at night are on the bottom.**

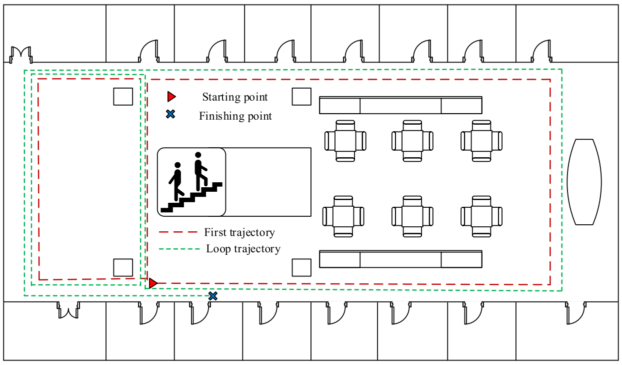


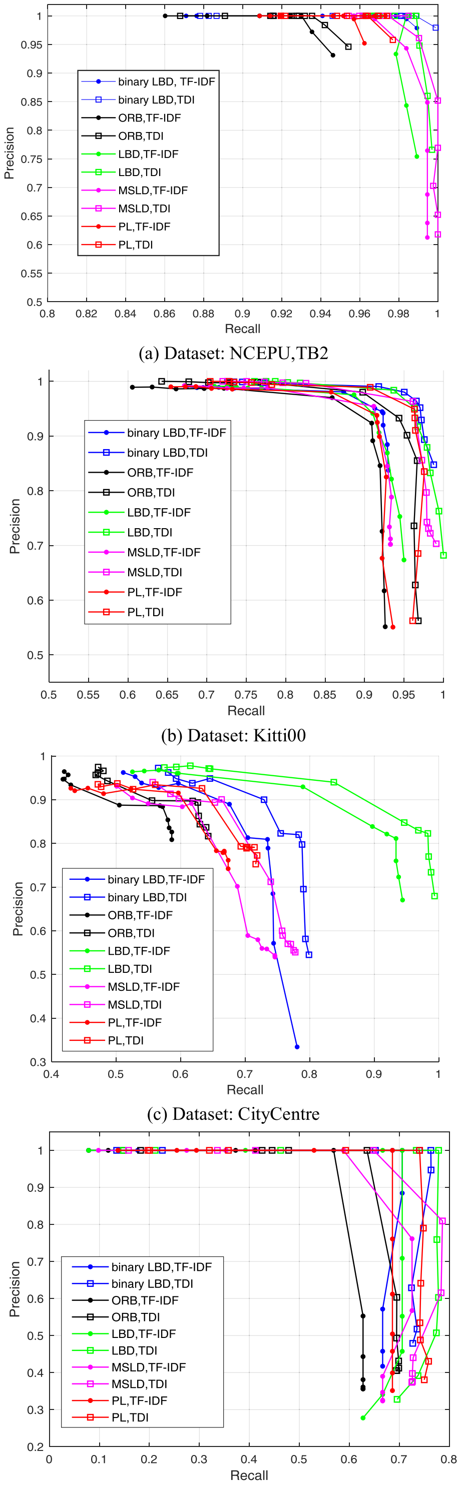
FIGURE 7. **A simple plan of the scene and the path robot followed.**

MSLD, LBD and PL-based methods, also with CNN-based loop closure detection method [41].

The datasets used in this section include two outdoor conditions (i.e. kitti00 sequence with 4541 frames and CityCentre with 1237 pairs images) and two indoor conditions (TUM, and a sequence collected in the 1st floor of NCEPU, No. 2 TB, with two loops, in total 240 m). Figure 7 shows a simple plans of the scene and the path robot followed to collect our image sequence.

Firstly, the Precision-Recall curves of loop closure detection results with different features are plotted as Figure 8 shows.

It can be seen that for all dataset and all features, the proposed TDI scheme outperforms TF-IDF scheme. For NCEPU, TB2 and kitti00 sequences, the performances of all features are compared as: binary LBD > LBD > MSLD > PL > ORB. Binary LBD descriptor achieves best performance. For CityCentre dataset, the comparison results are: LBD > binary LBD > MSLD > PL > ORB. LBD gives a very competent performance. For TUM dataset, the comparison results are also: LBD > binary LBD > MSLD > PL > ORB. Thus, LBD and binary LBD show better performances. But the performances of LBD, binary LBD, MSLD, and PL are very close. There are many similar scenes in these datasets, the results also verify that the proposed algorithm achieves competent performance in coping with perceptual aliasing conditions.



(a) Dataset: NCEPU,TB2

(b) Dataset: Kitti00

(c) Dataset: CityCentre

(d) Dataset: TUM

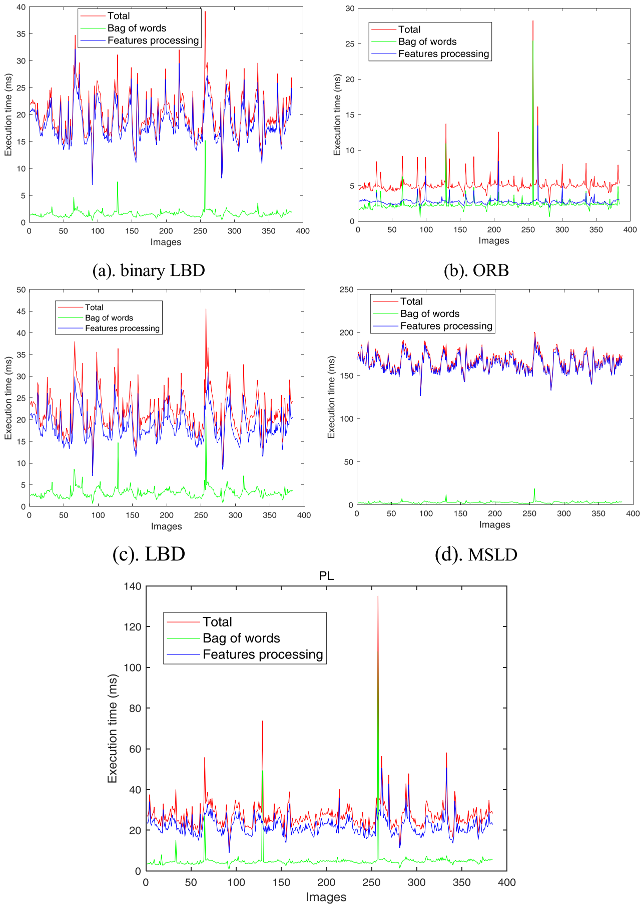
FIGURE 8. **Precision-recall curves achieved by ORB, MSLD, LBD, binary LBD and PL features using different dataset.**

TABLE 7. Results comparison.

Dataset		ORB	Binary LBD	LBD	MSLD	PL	CNN-based
TB2	P	0.9794	1	1	0.9612	1	0.9040
	R	0.9277	0.9808	0.9866	0.9903	0.9693	0.8124
kitti00	P	0.9327	0.9803	0.9517	0.8559	0.9490	0.8535
	R	0.9436	0.9503	0.9630	0.9726	0.9432	0.9226
City	P	0.8980	0.9003	0.9401	0.7122	0.9260	0.8313
	R	0.5556	0.7692	0.8373	0.7396	0.6763	0.7156
TUM	P	0.6032	0.8901	0.9826	0.8094	0.7900	0.7051
	R	0.6950	0.7701	0.7786	0.7863	0.7482	0.6869

In this experiment, it was consistent with [11] that, when α was 0.3, it achieved the best performance for all features. Hence, the precisions and recalls of all features with $\alpha = 0.3$ were compared with CNN-based method which proposed in [41]. Meanwhile, for the CNN-based method, we used the similarity threshold which received the best performance. Results are shown in Table.7, while 'P' represents 'precision', 'R' represents 'recall'. It can be seen that for NCEPU.TB2, kitti00 and TUM datasets, binary LBD feature obtains the best precisions, for CityCentre dataset, LBD achieves a higher value then binary LBD. For all datasets, the recalls of binary LBD are slightly lower than LBD and MSLD feature, but they are very close, which is still acceptable. The results of PL-based method is among the line only feature and point only based method, the reason is same as before description, because both point and line affect the similarity score that the result is weakened by points. The results of CNN-based method were achieved by setting the similarity threshold as: tb2_thr = 0.60, kitti00_thr = 0.60, city_thr = 0.35, tum_thr = 0.45, the best performance was obtained by set these values. The results are lower than feature-based method, it is because that temporal consistency check and spatial continuity check were not included to reject false loops. In addition, the feature of CNN-based method in [41] obtained is global descriptor that it is more susceptible to change in the viewpoint and scene occlusion.

Secondly, we also measured the execution time for each feature. Take NCEPU.TB2 dataset with $\alpha = 0.3$ as example, Figure 9 shows the execution time expended for each image with (a) binary LBD, (b) ORB, (c) LBD, (d) MSLD and (e) PL features. Table 8 reports the required time of feature processing and bag of words for experimental images. Feature processing time includes the time of feature detection, unaccepted features removing (e.g. the line length more than 20 pixels is accepted), and descriptors calculation. The time of bag of words includes: the time for transforming image into a bag of words vector, choosing candidate loop frames, constructing islands, selecting of the best candidate loop frame, loop decision and inserting current image into database.



(a). binary LBD

(b). ORB

(c). LBD

(d). MSLD

(e). PL

FIGURE 9. The execution time of loop closure detection with different features.

From Figure 9 we can see that the features processing step reveals higher execution time than bag of words step for line features. In aspect of feature processing time, among the line features, MSLD presents a far higher execution time (163.429ms/image) than LBD (18.3978ms/image) and binary LBD(18.5725ms/image) features. The execution time of binary LBD is slightly longer than LBD feature. It is because binary LBD descriptor is generated based on LBD descriptor, thus increases execution time. The execution time of PL is higher than LBD, binary LBD and ORB features, it is because that both orb and binary LBD were extracted. In aspect of the time of bag of words step, binary LBD shows the lowest execution time 1.5969 ms/query. Compared with MSLD and LBD, its binary descriptor helps to improve processing efficiency. Compared with ORB, ORB presents higher time with 2.3853 ms/ query. It is because the time of bag of words mainly relies on the number of features. In general, the number of extracted point features are more than line features for each image. PL shows the longest execution time 5.0013 ms/query. It is because both scores of points and lines need to compute. In aspect of whole loop execution time of each query, among line features, binary LBD shows a lowest time with 19.9947 ms/query. LBD presents a higher time than binary LBD. In contrast, MSLD gives an approximate 8 times of binary LBD execution time.

The efficiency of calculation is a key performance index for loop closure detection. With the best calculation efficiency, the proposed binary LBD based loop closure detection has an obvious advantage. In addition, LBD based loop closure

TABLE 8. Execution time with different features.

		Execution time(ms/ query)			
		Mean	Std	Min	Max
Features processing	binary LBD	18.5725	3.4381	6.9833	32.1306
	ORB	2.7534	0.7201	1.8900	13.4401
	LBD	18.3978	3.5265	7.0500	31.0626
	MSLD	163.4290	9.6552	126.4644	194.8188
	PL	22.1712	4.7652	8.8735	50.5872
Bag of words	binary LBD	1.5969	0.8894	0.3300	15.2076
	ORB	2.3853	1.3540	0.5347	25.4208
	LBD	3.2195	1.4750	0.7055	20.7371
	MSLD	2.8214	1.2624	0.4938	18.8197
	PL	5.0013	5.9342	1.0092	107.8062
Whole system	binary LBD	19.9947	3.9507	7.3133	39.1758
	ORB	5.1387	1.6167	2.4565	28.2527
	LBD	21.7920	4.5842	7.7555	45.5177
	MSLD	166.2504	10.3616	126.9582	200.2483
	PL	27.1725	8.1316	9.8827	135.0540

detection shows the best precision and recall rates. Therefore, users can select LBD or binary LBD based method, it depends on the practical demand.

VII. CONCLUSION AND FUTURE WORK

This work presents a novel loop closure detection method which applies only line features for the environment with structural elements, such as indoors, urbans, streets, etc. The bag of words model is extended in this loop closure detection algorithm to recognize revisited places. A variant of TF-IDF, TDI weighting scheme is proposed to improve the discrimination of similar score of two images, which contributes to improve the evaluation of two images' similarity. Experiments show that it enhances the discrimination of visual words, thereby improves the calculation accuracy of similarity. The binary LBD and LBD descriptors are used to construct visual vocabularies. The retrieval performance of these features are compared with ORB, MSLD and PL features using a few datasets. LBD vocabulary obtains the highest retrieval successful rate, binary LBD also works well. The performance of proposed whole loop closure detection algorithm is compared with ORB, MSLD, and PL-based loop closure detection methods and CNN-based method, results indicate that our method offers a very competent performance, it can cope with the perceptual aliasing conditions very well. In addition, binary LBD descriptor shows good ability in terms of calculation efficiency. To sum up, we can say that our algorithm offers a reliable loop closure detection result under man-made environment, which is beneficial to the vision-based SLAM. In the future work, we will investigate different type of CNN features and multiple descriptors based image representation approaches to extend the application scope of our work.

REFERENCES

[1] D. Valiente, A. Gil, L. Payá, J. M. Sebastián, and O. Reinoso, "Robust visual localization with dynamic uncertainty management in omnidirectional SLAM," *Appl. Sci.*, vol. 7, no. 12, p. 1294, 2017.

[2] K. L. Ho and P. Newman, "Detecting loop closure with scene sequences," *Int. J. Comput. Vis.*, vol. 74, no. 3, pp. 261–286, Sep. 2007.

[3] X. Gao and T. Zhang, "Unsupervised learning to detect loops using deep neural networks for visual SLAM system," *Auton. Robots*, vol. 41, no. 1, pp. 1–18, 2017.

[4] X. Wang, G. Peng, and H. Zhang, "Combining multiple image descriptions for loop closure detection," *J. Intell. Robotic Syst.*, vol. 92, nos. 3–4, pp. 565–585, 2018.

[5] D. Hahnel, W. Burgard, D. Fox, and S. Thrun, "An efficient fast SLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2003, pp. 206–211.

[6] H. Zhang, Y. Liu, and J. Tan, "Loop closing detection in RGB-D SLAM combining appearance and geometric constraints," *Sensors*, vol. 15, no. 6, pp. 14639–14660, 2015.

[7] M. Cummins and P. Newman, "Appearance-only SLAM at large scale with FAB-MAP 2.0," *Int. J. Robot. Res.*, vol. 30, no. 9, pp. 1100–1123, Jun. 2011.

[8] M. Labbé and F. Michaud, "Appearance-based loop closure detection for Online large-scale and long-term operation," *IEEE Trans. Robot.*, vol. 29, no. 3, pp. 734–745, Jun. 2013.

[9] Y. Latif, C. Cadena, and J. Neira, "Robust loop closing over time for pose graph SLAM," *Int. J. Robot. Res.*, vol. 32, no. 14, pp. 1611–1626, Oct. 2013.

[10] F. Amorós, L. Payá, J. M. Marín, and O. Reinoso, "Trajectory estimation and optimization through loop closure detection, using omnidirectional imaging and global-appearance descriptors," *Expert Syst. Appl.*, vol. 102, pp. 273–290, Jul. 2018.

[11] D. Galvez-López and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Trans. Robot.*, vol. 28, no. 5, pp. 1188–1197, Oct. 2012.

[12] H. Kwon, K. M. A. Yousef, and A. C. Kak, "Building 3D visual maps of interior space with a new hierarchical sensor fusion architecture," *Robotics Auton. Syst.*, vol. 61, pp. 749–767, Aug. 2013.

[13] N. Kejriwal, S. Kumar, and T. Shibata, "High performance loop closure detection using bag of word pairs," *Robot. Auton. Syst.*, vol. 77, pp. 55–65, Mar. 2016.

[14] L. Bampis, A. Amanatiadis, and A. Gasteratos, "Fast loop-closure detection using visual-word-vectors from image sequences," *Int. J. Robot. Res.*, vol. 37, no. 1, pp. 62–82, Jan. 2018.

[15] L. Bampis and A. Gasteratos, "Revisiting the bag-of-visual-words model: A hierarchical localization architecture for mobile systems," *Robot. Auton. Syst.*, vol. 113, pp. 104–119, Mar. 2019.

[16] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2006, pp. 2161–2168.

[17] Y. Li, Z. Hu, G. Huang, Z. Li, and M. A. Sotelo, "Image sequence matching using both holistic and local features for loop closure detection," *IEEE Access*, vol. 5, pp. 13835–13845, 2017.

[18] J. Sola, T. Vidal-Calleja, J. Civera, and J. M. M. Montiel, "Impact of landmark parametrization on monocular EKF-SLAM with points and lines," *Int. J. Comput. Vis.*, vol. 97, no. 3, pp. 339–368, 2012.

[19] T. Lemaire and S. Lacroix, "Monocular-vision based SLAM using line segments," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 2791–2796.

[20] M. Chandraker, J. Lim, and D. Kriegman, "Moving in stereo: Efficient structure and motion using lines," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep./Oct. 2009, pp. 1741–1748.

[21] G. Zhang and I. H. Suh, "A vertical and floor line-based monocular SLAM system for corridor environments," *Int. J. Control, Autom. Syst.*, vol. 10, pp. 547–557, Jun. 2012.

[22] J. O. Esparza-Jiménez, M. Devy, and J. L. Gordillo, "Visual EKF-SLAM from heterogeneous landmarks," *Sensors*, vol. 16, no. 4, p. 489, 2016.

[23] R. Gomez-Ojeda, D. Zuñiga-Noël, F.-A. Moreno, D. Scaramuzza, and J. Gonzalez-Jimenez, "PL-SLAM: A stereo SLAM system through the combination of points and line segments," 2017, *arXiv:1705.09479*. [Online]. Available: https://arxiv.org/abs/1705.09479

[24] Y. He, J. Zhao, Y. Guo, W. He, and K. Yuan, "PL-VIO: Tightly-coupled monocular visual–inertial odometry using point and line features," *Sensors*, vol. 18, no. 4, p. 1159, 2018.

[25] J. H. Lee, G. Zhang, J. Lim, and I. H. Suh, "Place recognition using straight lines for vision-based SLAM," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2013, pp. 3799–3806.

[26] Z. Wang, F. Wu, and Z. Hu, "MSLD: A robust descriptor for line matching," *Pattern Recognit.*, vol. 42, no. 5, pp. 941–953, 2009.

[27] L. Zhang and R. Koch, "An efficient and robust line segment matching approach based on LBD descriptor and pairwise geometric consistency," *J. Vis. Commun. Image Represent.*, vol. 24, no. 7, pp. 794–805, 2013.

[28] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: A survey," *IEEE Trans. Robot.*, vol. 32, no. 1, pp. 1–19, Feb. 2016.

[29] F. Han, H. Wang, G. Huang, and H. Zhang, "Sequence-based sparse optimization methods for long-term loop closure detection in visual SLAM," *Auton. Robots*, vol. 42, pp. 1323–1335, Oct. 2018.

[30] Y. Latif, G. Huang, J. J. Leonard, and J. Neira, "An online sparsity-cognizant loop-closure algorithm for visual navigation," in *Proc. Robots, Sci. Syst.*, Jul. 2014, pp. 1–9.

[31] Y. Liu and H. Zhang, "Visual loop closure detection with a compact image descriptor," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 1051–1056.

[32] M. J. Milford and G. F. Wyeth, "SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 1643–1649.

[33] B. Talbot, S. Garg, and M. Milford, "OpenSeqSLAM2.0: An open source toolbox for visual place recognition under changing conditions," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 7758–7765.

[34] A. Angeli, D. Filliat, S. Doncieux, and J.-A. Meyer, "Fast and incremental method for loop-closure detection using bags of visual words," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1027–1037, Oct. 2008.

[35] M. Cummins and P. Newman, "Probabilistic appearance based navigation and loop closing," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 2042–2048.

[36] M. Labbé and F. Michaud, "Online global loop closure detection for large-scale multi-session graph-based SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2014, pp. 2661–2666.

[37] Y. Hou, H. Zhang, and S. Zhou, "Convolutional neural network-based image representation for visual loop closure detection," in *Proc. IEEE Int. Conf. Inf. Autom.*, Aug. 2015, pp. 2238–2245.

[38] N. Sünderhauf, F. Dayoub, S. Shirazi, B. Upcroft, and M. Milford, "On the performance of convnet features for place recognition," 2015, *arXiv:1501.04158*. [Online]. Available: https://arxiv.org/abs/1501.04158

[39] N. Sünderhauf, S. Shirazi, A. Jacobson, F. Dayoub, E. Pepperell, B. U. Edward, and M. Milford, "Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free," in *Proc. 12th Robot., Sci. Syst.* 2015, pp. 1–10.

[40] R. Arroyo, P. F. Alcantarilla, L. M. Bergasa, and E. Romera, "Fusion and binarization of CNN features for robust topological localization across seasons," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 4656–4663.

[41] X. Zhang, Y. Su, and X. Zhu, "Loop closure detection for visual SLAM systems using convolutional neural network," in *Proc. 23rd Int. Conf. Autom. Comput. (ICAC)*, Sep. 2017, pp. 1–6.

[42] E. Sizikova, V. K. Singh, B. Georgescu, M. Halber, K. Ma, and T. Chen, "Enhancing place recognition using joint intensity—Depth analysis and synthetic data," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 901–908.

[43] X. Fei, K. Tsotsos, and S. Soatto, "A simple hierarchical pooling data structure for loop closure," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 321–337.

[44] P. Pinies, L. M. Paz, D. Galvez-Lopez, and J. D. Tardos, "CI-graph SLAM for 3D reconstruction of large and complex environments using a multicamera system," *J. Field Robot.*, vol. 27, no. 5, pp. 561–586, 2010.

[45] C. Cadena, D. Galvez-López, J. D. Tardos, and J. Neira, "Robust place recognition with stereo sequences," *IEEE Trans. Robot.*, vol. 28, no. 4, pp. 871–885, Aug. 2012.

[46] S. Oishi, Y. Inoue, J. Miura, and S. Tanaka, "SeqSLAM++: View-based robot localization and navigation," *Robot. Auton. Syst.*, vol. 112, pp. 13–21, Feb. 2019.

[47] R. G. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "LSD: A line segment detector," *IPOL J.*, vol. 2, pp. 35–55, Mar. 2012.

[48] J.-L. Blanco, F.-A. Moreno, and J. Gonzalez, "A collection of outdoor robotic datasets with centimeter-accuracy ground truth," *Auton. Robots*, vol. 27, no. 4, p. 327, Nov. 2009.

[49] *Indoor Level 7 S-Block Dataset*. Accessed: Jan. 1, 2018. [Online]. Available: https://wiki.qut.edu.au/display/cyphy/Indoor+Level+7+S-Block+Dataset

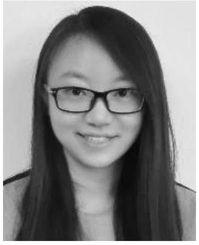
[50] M. Smith, I. Baldwin, W. Churchill, R. Paul, and P. Newman, "The new college vision and laser data set," *Int. J. Robot. Res.*, vol. 28, no. 5, pp. 595–599, May 2009.

[51] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, and R. Siegwart, "The EuRoC micro aerial vehicle datasets," *Int. J. Robot. Res.*, vol. 35, no. 10, pp. 1157–1163, 2016.

[52] J. Fritsch, T. Küehnl, and A. Geiger, "A new performance measure and evaluation benchmark for road detection algorithms," in *Proc. 16th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2013, pp. 1693–1700.

[53] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 573–580.

[54] Dataset1. *Hanyang University*. Accessed: Jan. 1, 2018. [Online]. Available: https://drive.google.com/file/d/0B3bB8rHbc3fWNGQ3YnhLczIwNG8/view?usp=sharing



RUIFANG DONG received the B.S. degree in automation and the M.S. and Ph.D. degrees in pattern recognition and intelligent system from North China Electric Power University, Beijing, China, in 2010, 2013, and 2018, respectively. She is currently a Lecturer with the School of Technology, Beijing Forestry University, China. Her research interests include computer vision, visual SLAM, and view planning.

ZHAN-GUO WEI is currently an Associate Professor with the Central South University of Forestry and Technology, Changsha, China. His research interests include logistics equipment, mobile robot, and artificial intelligence.

CHANG-AN LIU received the B.S. degree from Northeast Agricultural University, in 1995, and the M.S. and Ph.D. degrees from the Harbin Institute of Technology, in 1997 and 2001, respectively. He is currently a Professor and the Director of the Intelligence Robot Institute, North China Electric Power University. His research interests include technology of intelligent robot and theory of artificial intelligence.

JIANGMING KAN is currently a Professor with the School of Technology, Beijing Forestry University, China. His research interests include forestry robot, computer vision, and image process.