Improved SLAM closed-loop detection algorithm based on DBoW2

Qingsong Zhang¹, Guanghui Xu*, Na Li¹
¹College of Communication Engineering, Army University of Science and Technology, Nanjing, Jiangsu, 210007, China
*Corresponding author’s e-mail: 13814060352@139.com

Abstract. In order to solve the problems of dbow2-based closed-loop detection algorithm, it is necessary to determine the appropriate level of dictionary tree and K value of clustering algorithm according to experience in advance. First, the new algorithm obtains the initial image feature clustering of training set by k-means ++ algorithm within the search range of cluster number (H, L). Secondly, by merging the form of similar classes, the cluster number is gradually reduced until the CRI function converges. Finally, the clustering value at this point is the best clustering value adopted by the training set to train the dictionary tree, and the clustering is repeated until the complete dictionary tree is generated. The feasibility of this algorithm is verified through experimental analysis, and the new method proposed can solve the problem of dictionary tree generation of training sets under different backgrounds, which plays a very important role in closed-loop detection based on visual image similarity detection, and has certain theoretical and practical reference value.

1. Introduction
In recent years, the closed-loop detection algorithm based on image similarity is an important part of SLAM problems studied at home and abroad. Many scholars have proposed a series of solutions. Literature [3] proposed to use the similarity between image and map to determine the consistency of image points and map points, but this method has a high computational complexity and is only applicable to closed-loop detection of global maps in a small range of environments. In the context of a wide range of environments, literature [4] proposed a rapid graph construction algorithm based on significance to achieve closed-loop detection. Images acquired online by the robot were used to build a database, and newly acquired images were judged to form a closed-loop by rapid matching with the database. In recent years, the BoW (Bag of Words) method proposed in literature [5] has become the main method for closed-loop detection of SLAM systems by virtue of its efficient and fast image matching. This method firstly clusters the data set composed of the image features of the training set and finally uses the visual words mapped by the cluster center to detect the similarity of the two images. However, this method is limited by the speed of image feature extraction, and its feature extraction cycle is about 10 times longer than other steps.

Literature [6] - [7] proposed the ORB - SLAM2 (Oriented FAST and Rotated in the SLAM 2) model based on DBoW2 (Bags of Binary Words for FAST Recognition in Image Sequence) algorithm for detecting the backend closed-loop. DBoW2 algorithm using the ORB (Oriented FAST and Rotated BRIEF) to extract the feature points from training set and generate the corresponding character description, and then through the K - means++ (K means clustering algorithm + +) clustering algorithm to generate layers, the number of clustering for the training of the K dictionary tree to calculate the
similarity between the image quickly, greatly improve the speed of feature point extraction. However, this method needs to determine the appropriate level of dictionary tree (set as 6 in the literature) and the value of clustering algorithm (set as 10 in the literature) in advance according to experience, and it is difficult to determine the appropriate value of different training sets in different environments. In addition, the clustering algorithm used by DBoW2 can achieve good clustering results for large data sets that are distributed in clusters, but it cannot achieve the best effect when faced with the ORB feature description operator set extracted from the image set of training set.

As an important part of the most mature and perfect feature-based real-time SLAM system, the closed-loop detection algorithm based on DBoW2 must be improved to improve the robustness of the system. In view of the above situation, this paper proposes a new optimal clustering number determination algorithm based on DBoW2’s original clustering method, determines the searching range of clustering number, and performs local value optimization on the number K of leaf nodes in each layer of the dictionary tree. Through theoretical demonstration and experiment, the new method proposed in this paper has better performance and feasibility.

2. Related work

2.1. Closed loop detection algorithm model based on DBoW2

Under the background of large-scale unknown environment, closed-loop detection plays an important role in realizing the robustness of SLAM system. After a period of time, when the robot moves to the position where it once appeared, due to the accumulation of unavoidable data errors after a long period of movement, it is difficult to achieve a true closed loop of its movement trajectory, and this problem is particularly prominent in large-scale complex environment. The correct closed-loop detection can detect whether the current position is closed at all times, and eliminate the accumulation of errors by closing the loop, so as to achieve the goal of optimizing the entire motion trajectory and global map.

Before many of the closed-loop detection model, adopting the SIFT (Scale Invariant Feature Transform) or SURF (Speeded Up Robust Features), such as Scale invariance Feature extraction algorithm. This is because they are largely unaffected by changes in lighting, scale and rotation, and can perform well even when the robot's perspective changes slightly. However, it has been proved by practice that these models are limited by feature extraction algorithms and the long period of image similarity detection, and cannot meet the real-time requirements in a large unknown environment. ORB-SLAM model, as one of the most mature and perfect feature-based real-time SLAM systems, adopts dbow2-based closed-loop detection algorithm. The effect of the model in KITTI Odometry Sequence-00 is shown in figure 1.

Fig.1 ORB_SALM2's running diagram on the KITTI dataset

DBoW2 algorithm using a BRIEF description of the generated containing 256 bit binary number vector to represent the image of the feature points, then according to the clustering strategy of K_means + + will these characteristics describe clustering into word, use the training set generated by the dictionary tree clustering words and images can be converted to its weight vector, by comparing the
hamming distance of binary vector images. And thanks to the hierarchical clustering structure of the
dictionary tree, each layer only needs K comparisons, and its search efficiency is greatly improved. The
closed-loop detection algorithm based on DBoW2 greatly improves the speed of similarity detection
and meets the real-time requirements of the system. Its structural framework is shown in figure 2.

![Fig.2 closed-loop detection model based on DBoW2 algorithm](image)

2.2. Dictionary tree clustering algorithm
The dictionary tree is trained by mapping the training set image to binary numerical vector space, which
reduces the speed of image similarity detection. A dictionary tree consists of a large number of visual
words with different feature points, all of which are combinations of certain kinds of features. Therefore,
the lexicon tree generation problem can be summarized as a clustering problem, which can be solved by
k-means ++ clustering algorithm. K-means++ algorithm is based on the improvement of k-means.
Different from the random generation of initial center points in the original version, the new algorithm
selects K clustering center points as far away from each other as possible through special strategies, so
as to make the clustering effect free from the influence of initial cluster center to the greatest extent.

In order to get enough visual words, the dictionary tree in DBoW2 algorithm adopts a layer-by-layer
clustering method: K is artificially determined and K clustering clusters are obtained by the clustering
algorithm; Then train each cluster again, and generate K cluster centers as well; Repeated $L_w$ times
will generate a dictionary tree $K - L_w$, and the leaves of the tree are the $K^{L_w-1}$ visual words that can be
trained to get up to one word. The specific implementation steps of the algorithm are as follows:

1) determine the initial cluster center value $K$ and the maximum depth of the dictionary tree $L_w$.
The feature descriptor set $\{ x_i | i = 1, 2, \cdots, N \}$ is obtained by extracting the features of the visual image
training set.

2) randomly select the first clustering center point in the feature description set, and calculate the
shortest distance from other points to the clustering center $H(i)$.

3) randomly select the next cluster center among the remaining points, and require the probability of
the remaining points to be selected to be allocated according to equation (1). Repeat step 3 until K
clustering centers are selected.
$$P(i) = \frac{H(i)}{\sum_{i=1}^{N} H(i)}$$ \tag{1}

4) Calculate the shortest distance from each point in the training set to the clustering center and divide it into the nearest clustering center to generate data cluster \( \{ C_i | i = 1, 2, \ldots, K \} \).

5) For the data cluster, the clustering center of the data cluster is recalculated according to formula (2), and step 4 is repeated until the center position of the data cluster is no longer changed. At this point, the first layer of dictionary tree structure is completed.

$$C_i = \frac{\sum x_i}{n}$$ \tag{2}

6) Repeat steps 2 through 5 in the existing data cluster until \( L_w \), the sub-clustering is completed to generate a complete dictionary tree \( K - L_w \), which generates a leaf node \( K^{L_w-1} \).

The orb-slam model extracts 9 million features from the independent data set of Bovisa 2008-09-01, adopts the 6-layer and 10-branch dictionary tree structure, so that the number of words generated reaches millions. Compared with the traditional word dictionary of 1000 words space, the dictionary tree has stronger representation ability.

However, this method needs to determine the level of dictionary tree and clustering K value according to experience in advance, and it is difficult to determine the empirical value of different training sets. In addition, DBoW2 can achieve a good clustering of cluster data by using clustering algorithm, but it obviously cannot achieve the best effect when facing the feature description operator set from the training set. Literature [7] proposed to use the idea of neighbor propagation to generate the upper search range of cluster number, and to determine the best cluster number through index analysis. However, this algorithm often produces too many clusters and can't give an accurate result when it comes to a relatively loose cluster structure. In this paper, in the process of training dictionary tree clustering, the self-adaptive adjustment of clustering K value is adopted to prevent the clustering process of dictionary tree from being affected by different training sets, so as to achieve the best clustering effect and ensure the accuracy of subsequent closed-loop detection modules. Firstly, the larger K value is selected as the initial clustering value of the dictionary tree, and then, on this basis, the new K value and the new clustering center are adjusted continuously by combining the generated clusters with high similarity.

3. The determination of the dictionary tree clustering value and the local optimal value based on the search range

As one of the most widely used clustering algorithms, k-means must set specific clustering value in advance, and inaccurate value will seriously affect the final effect of clustering. In addition, in the face of complex clustering data structure, the clustering effect is easily affected by the clustering center, leading to a large difference in clustering results. In this case, the new clustering algorithm USES the given maximum clustering value and k-means ++ algorithm to get the initial training set image feature clustering. Then, by combining the form of similar classes, the clustering number is gradually reduced until the evaluation value designed in this paper converges. At this time, the clustering value is the optimal clustering K value that should be adopted by the training dictionary tree of the corresponding training set.

3.1. Theoretical analysis of the improved algorithm
When defining intra-cluster compactness of data clusters, intra-class aggregation in S_Dbwn index in literature [4] was introduced to evaluate compactness. The compactness within the cluster, where the cluster number is, is defined as follows:

$$CN(k) = \frac{1}{k} \sum_{i=1}^{k} \frac{\sigma(C_i)}{\sigma(C)}$$  \hspace{1cm} (3)

Where, $\sigma(C_i)$ is the variance of data within the cluster of the $i$ data cluster, and $\sigma(C)$ is the total variance of all clusters. The smaller the value of $CN(k)$, the stronger the aggregation of the whole data cluster, the better the clustering effect of the algorithm.

However, the dispersion function in S_Dbwn index is always calculated by drawing circles, so the optimal clustering effect cannot be achieved when dealing with non-spherical data clusters. However, the feature descriptors collected from the image training set by the dictionary tree do not have spherical features. In order to better represent the dispersion of data clusters, this paper adopts the nearest neighbor based dispersion function between clusters. If an object is located in the center of a data cluster and surrounded by objects of the same data cluster, it is well isolated from other data clusters and contributes little to the dispersion estimation among clusters. If an object is located at the edge of the data cluster and is mainly surrounded by objects in other data clusters, then it will be associated with other cluster groups and occupy a relatively important position in the measurement of inter-cluster dispersion.

According to the idea of nearest neighbor, the dispersion of data clusters is defined as follows:

$$CJ(k) = \max_{i=1,2,3,...,n} \left( \frac{1}{n_i} \sum_{j=1}^{n_i} \left( \frac{q_j}{C} \right) \right)$$  \hspace{1cm} (4)

Where, $n_i$ represents the number of data $i$ in the cluster of the first data cluster, $q_j$ represents the number of data in other data clusters of adjacent points, and $C$ represents the total number of adjacent points. The smaller the value of $CJ(k)$, the stronger the aggregation of the whole data cluster, the better the clustering effect of the algorithm.

The nature of data set classification is not a completely solved problem, and the importance of compactness and dispersion varies according to the application field. DBoW2 uses ORB algorithm to extract the image descriptor of training set, and then clusters it into words according to the clustering strategy, which is detected by comparing the hamming distance of binary vector. In order to achieve better similarity detection based on image feature points dispersion obviously plays a more important role. The new algorithm reduces the cluster number gradually by combining the form of similar classes until the evaluation value designed in this paper converges. At this time, the clustering value is the optimal clustering K value that should be adopted by the training dictionary tree of the corresponding training set. According to the compactness and dispersion functions of data clusters, the convergence function is defined as:

$$CRI = \frac{CN(k) - CN(k-1)}{CJ(k) - CJ(k-1)} = \frac{\Delta CN(k)}{\Delta CJ(k)}$$  \hspace{1cm} (5)

Where, $CN(k)$, $CJ(k)$ is the compactness and dispersion function before the merger, and $CN(k-1)$, $CJ(k-1)$ is the function after the merger. In the new algorithm proposed in this paper, select (H,L) as the search range to find the best clustering value. The faster the increase rate is $CN(k)$, the less compact the data in the merged data cluster will be, and the objects in the cluster will be more scattered. On the contrary, it means that the data clusters before merging are more compact, so the smaller $CN(k)$ the increase, $CJ(k)$ the better. Similarly, the faster the growth rate before and after the merger, the greater the dispersion of data clusters will be, and the better the final classification effect will be.

By comparing the amplitude changes of the compactness and dispersion functions in equation (5), the increasing speed $CN(k)$ is fast, so the value of $CRI$ is smaller and the clustering effect is better.
Otherwise, the higher the value of $CRI$, the faster the compactness grows than the dispersion, and the worse the clustering effect will be.

3.2. Improved algorithm implementation steps

Since the training set image descriptor of the dictionary tree is a 256-dimensional binary feature, and the hamming distance is used to calculate the distance between the feature vectors representing the image in the similarity detection, the final discrete distance value is at most 256. Therefore, in the clustering algorithm of the dictionary tree, the clustering value should not be too high, but the number of words contained in the dictionary tree must reach enough order of magnitude to carry out efficient image detection. Therefore, in the new algorithm proposed in this paper, select $(H,L)$ as the search range to find the best clustering value, and search for the best clustering effect locally. In this paper, $(13,8)$ is selected as the possible range of clustering value, and is specified as 6 to carry out dictionary tree feature clustering. Specific algorithm implementation steps are as follows:

1) The feature descriptor set $\{x_i|i=1,2,\ldots,N\}$ is obtained by extracting the features of the visual image training set.

2) K-means ++ algorithm is used to get the initial image feature clustering of training set, and H clustering centers are generated.

3) The dispersion $CN(k)$ and compactness functions $CJ(k)$ of the cluster are calculated.

4) For the H data cluster centers obtained, the two closest data clusters are merged, and the current cluster center is calculated.

5) Calculate the shortest distance from each point in the training set to the clustering center and divide it into the nearest clustering center to generate data cluster $\{C_i|i=1,2,\ldots,H-1\}$.

6) For the data cluster $C_i$, the clustering center of the data cluster is recalculated according to equation (2), and step 5 is repeated until the center location of the data cluster is no longer changed.

7) Calculate $CN(k-1)$ and $CJ(k-1)$.

8) Calculate the function $CRI$ to determine whether the current merge is reasonable. If it is reasonable to repeat steps 2 through 8 until the cluster value is L after the merge, otherwise return the cluster value before the merge and proceed to the next step.

9) Get the current clustering value, and use the new clustering center to complete k-means ++ clustering. At this point, the first layer of dictionary tree structure is completed.

10) In the data clusters that have been classified, repeat step 9 with the best clustering value obtained at present until $L_w$ sub-clustering is completed, thus generating a complete dictionary tree $K-L_w$ and generating a leaf node $K^{L_w-1}$.

4. Experimental verification

This experiment USES Intel(R) 3.60GHz, 8.00G memory software environment, and Windows 7 and Ubuntu14.04 dual system environment for development and implementation. The new improved algorithm chooses $(H,L)$ as the searching range and seeks for the best clustering value locally. However, the algorithm is applicable to any data set of clustering number, and the implementation principle is the same. Moreover, the experiment of KITTI data set Odometry Sequence-00 is performed by the improved algorithm under Ubuntu14.04 environment.

In order to quickly verify the correctness of the improved algorithm, three data sets in UCI database, namely the classic Iris data set, the Wine data set and the synthetic data set, were clustering analyzed. Since the image descriptor of the training set of the dictionary tree is a 256 dimensional binary feature, the artificial data set is composed of 256 dimensional data composed of 7 centers in order to be closer to the real environment. The specific situation of the three data sets is shown in table 1. UCI standard data set is a database specially built for machine learning and data mining research. It is convenient for relevant research, and the correctness of experimental results can be verified by comparing rich and best-classified data sets in the database.
Table 1 Specific properties of different data sets

| Datesheet | Clustering number | source | Number of samples | Number of sample categories |
|-----------|------------------|--------|-------------------|-----------------------------|
| Iris      | 3                | UCI    | 150               | 4                           |
| Wine      | 4                | UCI    | 178               | 14                          |
| Datesheet 3 | 7              | artificial | 800            | 256                         |

Experimental analysis was carried out on the three data sets of Iris, Wine and artificial synthesis, and the experimental results were shown in table 2. In order to see the function changes more clearly, the broken line graph of CRI function was given, as shown in figure 4.

Table 2 simulation values of CRI function

| Cluster number after merging | Iris     | Wine     | Datesheet 3 |
|-----------------------------|----------|----------|-------------|
| 12                          | 8.537    | 9.654    | 14.356      |
| 11                          | 4.385    | 8.245    | 38.256      |
| 10                          | 8.264    | 12.310   | 25.325      |
| 9                           | 6.435    | 4.568    | 12.589      |
| 8                           | 5.324    | 3.754    | 20.365      |
| 7                           | 3.257    | 7.568    | 21.365      |
| 6                           | 7.431    | 10.356   | 256.654     |
| 5                           | 12.384   | 4.564    | 89.356      |
| 4                           | 9.365    | 8.378    | 168.654     |
| 3                           | 6.384    | 38.539   | 124.365     |
| 2                           | 35.28    | 56.354   | 56.148      |

Fig. 3 Line graph of CRI function changing with K value

It can be seen from figure 3 that the function value will experience a rapid rise stage after the clustering reaches the optimal clustering number. At this time, due to the reduction of the number of clusters, the compactness function of the data will increase rapidly, while the dispersion function of the data will decrease, and the clustering effect will decline rapidly. Therefore, it can be concluded that the new algorithm can get the correct dictionary tree clustering value, and it is also suitable for high latitude data sets. Moreover, experiments of KITTI data set Odometry Sequence-00 were performed in Ubuntu environment, and good results were obtained, as shown in Figure 4.
5. Conclusion

To solve the problems of the dbow2-based closed-loop detection algorithm, such as the number of appropriate dictionary tree layers and the K value of the clustering algorithm, a dictionary tree generation algorithm which can be applied to different training sets is proposed. In the search range of the best clustering value, the new algorithm obtains the initial image feature clustering of training set by k-means ++ algorithm. Then, the clustering number is gradually reduced by merging similar classes. Finally, the CRI function based on data compactness and dispersion is used to obtain the best clustering number of the generated dictionary tree, and finally the complete dictionary tree is generated. In addition, the feasibility of the proposed algorithm is verified by experimental analysis. The new algorithm plays a very important role in the closed loop detection based on the similarity detection of visual images and has certain theoretical and practical reference value.

References

[1] Leonard, J., Durrant-Whyte, H. (1991) Mobile Robot Localization by Tracking Geometric Beacons. J. IEEE Transactions on Robotics and Automation., 7:376-382.
[2] Davide, S., Friedrich, F. (2011) Visual Odometry, Part I: The First 30 Years and Fundamentals. J. IEEE Robotics & Automation Magazine., 2011:80-92.
[3] Faugeras O.D., Lustnian, F. (1988) Motion and structure from motion in a piecewise planar environment. J. International Journal of Pattern Recognition and Artificial Intelligence., 2: 485-508.
[4] Hartley, R., Zisserman, A. (2003) Multiple view geometry in computer vision. Cambridge university press, America.
[5] Mur-Artal, R., Montiel, M., Tardos, J.D. (2015) ORB_SLAB: A versatile and accurate monocular SLAM system. J. IEEE Transactions on Robotics., 31: 1147-1163.
[6] Klein, G., Murray, D. (2007) Parallel tracking and mapping for small AR workspaces CI/Mixed and Augmented Reality. In: 6th IEEE and ACM International Symposium. America. pp. 225-234.
[7] Strasdat, H., Davison, A.J. (2012) Visual SLAM: why filter? J. Image and Vision., 30: 65-77.