

A Brain-inspired SLAM System Based on ORB Features

Sun-Chun Zhou Rui Yan Jia-Xin Li Ying-Ke Chen Huajin Tang

Neuromorphic Computing Research Center, College of Computer Science, Sichuan University, Chengdu 610065, China

Abstract: This paper describes a brain-inspired simultaneous localization and mapping (SLAM) system using oriented features from accelerated segment test and rotated binary robust independent elementary (ORB) features of RGB (red, green, blue) sensor for a mobile robot. The core SLAM system, dubbed RatSLAM, can construct a cognitive map using information of raw odometry and visual scenes in the path traveled. Different from existing RatSLAM system which only uses a simple vector to represent features of visual image, in this paper, we employ an efficient and very fast descriptor method, called ORB, to extract features from RGB images. Experiments show that these features are suitable to recognize the sequences of familiar visual scenes. Thus, while loop closure errors are detected, the descriptive features will help to modify the pose estimation by driving loop closure and localization in a map correction algorithm. Efficiency and robustness of our method are also demonstrated by comparing with different visual processing algorithms.

Keywords: Simultaneous localization and mapping (SLAM), RatSLAM, mobile robot, oriented features from accelerated segment test and rotated binary robust independent elementary (ORB) features of RGB (red, green, blue), cognitive map.

1 Introduction

Animals have an instinctive ability to explore and navigate in an unknown space. Inspired by the spatial cognition of animals, in the past decades, many researchers were investigating how animals perceive, store and maintain spatial knowledge^[1–7]. Tolman^[1] first thought that cognitive map may exist in rodent's brain in 1948. In 1971, O'Keefe and Dostrovsky^[2] found place cells, located in hippocampus, increase their firing rate when rodent is in a particular position in a maze. With the development of neurology, neurophysiological studies^[3–6] then revealed that entorhinal cortex (EC) has an internal representation of the environment by spatially encoded cells identified as "grid cells". These cells are activated in a hexagonal pattern during spatial navigation and are also independent of external environment. Furthermore, experimental results^[7] evidenced that hippocampus can receive all sensory neocortical information from EC due to connections between EC and hippocampus. These biological findings motivated researchers in robotics area to build a cognitive map by a process integrating activity from both the grid cells and place cells in the EC-hippocampal area^[8–11].

These findings not only help us have a deeper understanding of brain "GPS" (global positioning system)^[12], but also trigger research on brain-inspired simultaneous localization and mapping (SLAM) and relevant

algorithms^[13–16]. Though the mechanism of brain "GPS" still has many unknown factors, it can be safely claimed that the SLAM system of rodents is not only based on internal path integration, but also involves external visual cues of physical world. Rodent can update the pose by fusing the external visual information and the neural activities of place cells and grid cells. These behaviors are similar to the prediction and update process in SLAM. Different from the existing SLAM algorithms based on probabilistic methods^[17, 18], RatSLAM^[19], a brain-inspired SLAM algorithm, creates a cognitive map based on visual information and the information of its previous experiences. RatSLAM simulates the rodent's spatial cognition mechanism using pose cells, local view cells and cognitive maps. Pose cells construct a continuous attractor network (CAN) which is suggested to demonstrate the path integration for grid cells in EC, local view cells functionally replace rodent's perceptual system and the cognitive map is an analogy of place cells in the hippocampus. In practice, Milford and Wyeth^[19] successfully implemented a coherent map after mapping a 66 km suburb using only a single web camera. However, the vision algorithm of this method is not fit for an office environment with many similar scenes. In order to map an office environment, Tian et al.^[20, 21] took advantage of RGB-D (red, green, blue and depth) information to construct a spatial cognitive map of an office environment. Similarly, the above methods just simply extract intensity scanline profiles from visual information and make a comparison among these scanline profiles.

In past years, various computer vision technologies have been proposed to detect and describe local features, which are widely applied in visual SLAM algorithms^[22–24].

Research Article
Special Issue on Human-inspired Computing
Manuscript received December 26, 2016; accepted April 14, 2017;
published online August 18, 2017
This work was supported by National Natural Science Foundation
of China (No. 61673283).
Recommended by Associate Editor Hong Qiao
© The Author(s) 2017

Scale-invariant feature transform (SIFT)^[25] is a popular and useful visual processing algorithm, which is highly invariant to image scaling and rotation, and even partially invariant to the illumination. Even over a decade, SIFT has been successfully applied in a set of applications. However, SIFT is not suitable for real time demand, because it takes massive time for calculating the scale factor. Therefore, many methods aimed at speeding up the computation of SIFT have been published^[26–32].

In particular, features from accelerated segment test (FAST)^[28] and binary robust independent elementary features (BRIEF)^[30] are two arguably outstanding methods. FAST uses machine learning to derive corner detector and provides a suitable way to finding key points in real-time systems, such as, parallel tracking and mapping (PTAM)^[33]. BRIEF is a binary feature descriptor that relies on a relatively small number of intensity difference tests to represent an image patch. One great advantage of BRIEF descriptors is that they are very fast to compute and to compare. Gálvez-López and Tardos^[34] proposed a real-time visual place recognition method by both FAST and BRIEF. Furthermore, by integrating FAST and BRIEF, Rublee et al.^[35] developed a visual processing method named oriented FAST and rotated BRIEF (ORB), which is an efficient alternative to SIFT. ORB uses FAST algorithm to detect key points in images, filters the key features using Harris corner measure^[36] and computes binary descriptors using oriented BRIEF. ORB can provide good performance with low cost computation, it is suitable to construct a semi-dense map in real-time^[37].

Inspired by these implementations of ORB and RatSLAM, ORB is adopted to replace RatSLAM algorithm's vision processing method in this paper. By fusion of RGB information and raw odometry information, a cognitive map can be built. This map contains a set of spatial coordinates that robot has traveled. In order to fully verify the effectiveness of our method, we also make a comparison with other visual processing method below. The experiment results show that ORB can significantly enhance robustness of the proposed SLAM system.

This paper is organized as follows. Section 1 has introduced the background of brain-inspired SLAM and discussed the purpose of this paper. An overview is given in Section 2 to describe system architecture and robot platform. In Section 3, visual processing method is explained in detail. The cognitive model of brain-inspired SLAM is described in Section 4. Experimental results are demonstrated in Section 5 to show the effectiveness of our proposed method. Finally, Section 6 gives a conclusion.

2 System overview

Fig. 1 shows the hardware architecture of robot in experiment. The robot consists of a mobile base, a RGB-D camera and a miniPC. We use Pioneer 3-DX as mobile base. The mobile base is a compact differential-drive mobile robot,

which includes a motion controller with built-in encoders. The robot's embedded motion controller performs velocity control of the robot and offers odometry information. A RGB-D camera is mounted on mobile base to capture visual information of environment. The miniPC is used to process visual information, odometry information and cognitive model.

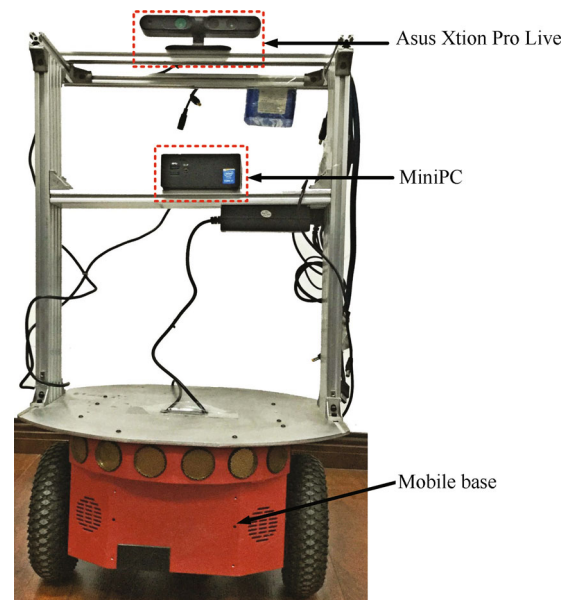


Fig. 1 Robot platform

A schematic overview of the cognitive map building system is shown in Fig. 2. The system mainly consists of visual processing module and cognitive map model. The visual processing module is responsible for extracting ORB features. In cognitive map model, the ORB features are associated to local view cells which represent existing cues at specific location. Raw odometry information from mobile base is involved in path integration process. The internal CAN dynamics ensures cognitive map model to converge at a steady state even without external sensors. Eventually, local view cells activity, path integration process and CAN dynamics work together to build a cognitive map.

3 Visual processing

Visual processing is an irreplaceable part of our SLAM algorithm. During mapping process, visual information is adopted to determine the generation of local view cells and eliminate redundant images; while in global localization process, robot captures incoming frames and compares them with recorded visual templates to determine its location. In previous implementation, Milford and Wyeth^[19] built a coherent map on large suburban areas. Milford's vision algorithm adopts a scanline intensity profile to deduce the translation and rotation with previously seen visual information. The scanline intensity profile is a one-dimensional vector by summing pixels in column and then

normalizing the vector. However, this processing way is imprecise and not a good fit for an office environment with many similar scenes. In this paper, we use ORB algorithm to replace the simple vision algorithm. The method is proved to be well-performing in experiments which will be illustrated in Section 5.

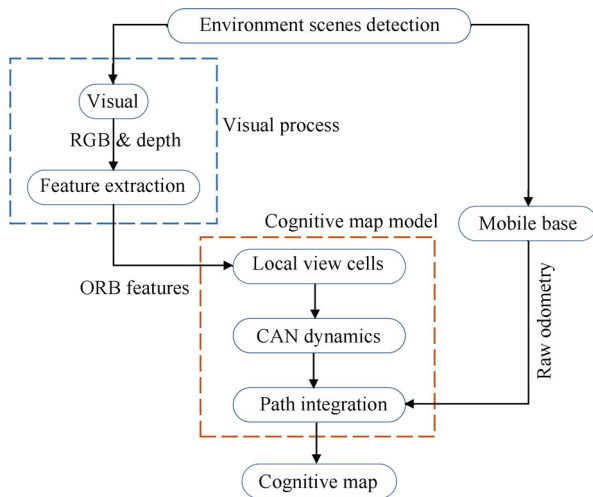


Fig. 2 System architecture

ORB is a combination of FAST detector and oriented BRIEF descriptor with modifications to enhance the performance. As the classical visual processing algorithm, FAST is very useful to detect key points. However, FAST cannot

produce multi-scale features and does not have an orientation component. To solve the former limitation, a scale pyramid of the image was employed to generate FAST features at each level in the pyramid. For latter limitation, the major improvement over FAST is that the intensity centroid is introduced to measure corner orientation. As for descriptor, BRIEF is one of the fastest available feature descriptors for calculation and matching, however it is variant to in-plane rotation. In order to address this limitation, Rublee adopts a steered BRIEF to improve the performance. The BRIEF descriptors in ORB should have two properties: lower correlation and high variance among binary tests. Low correlation among binary tests indicates each new test will bring new information into the descriptors, thus maximizing the amount of information recorded in descriptor. High variance among binary tests makes features of image more discriminative. Rublee developed a learning method to ensure descriptors have these two properties. This method searches for a set of uncorrelated binary tests in an image patch, as shown in Fig. 3.

Fig. 3 presents a flow chart of the learning method in ORB to choose a good set of binary features with high variance and low correlation. In this method, all possible binary tests in an image patch are firstly ordered to obtain high variance binary tests. After that, a greedy search process is implemented to filter the high correlation binary tests. Finally, the selected binary tests form an ORB descriptor.

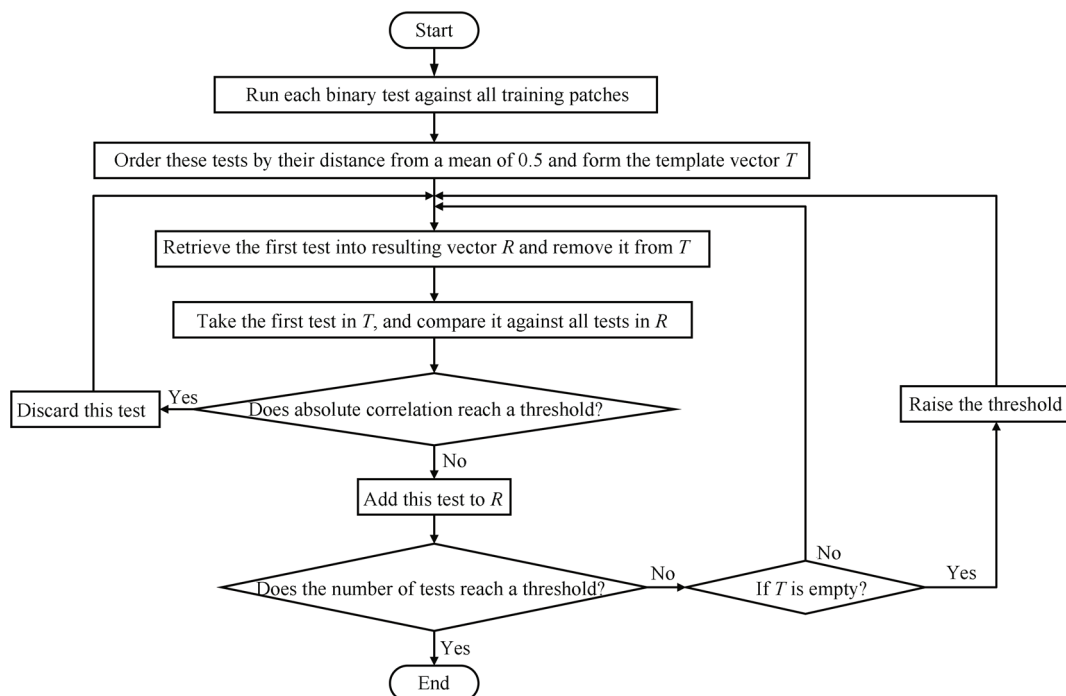


Fig. 3 Flow chart of learning method to choose a good set of binary features with high variance and low correlation

By the integration of FAST and BRIEF approaches, ORB method has been proposed. Experimental results showed that for the same number of features, the processing time of ORB is faster than speeded up robust features (SURF) by an order of magnitude and over two orders faster than SIFT. ORB was used to achieve the good performance for place recognition^[37, 38]. Comparing with the other feature descriptions such as SIFT, SURF and FAST, major advantages of ORB include: 1) It provides a fast calculation and matching algorithm. 2) ORB retains the ability of invariance to image scaling and rotation. 3) ORB is relatively immune to image noise, lighting and perspective distortion.

Thus, ORB method is employed in this paper to extract features from images as visual template. When the current image matches prior visual templates, it is considered that robot reached this place previously. Otherwise, a new visual template is added to local view cells, as shown in Fig. 4.

4 Cognitive map model

The cognitive map model mainly contains several components: local view cells, CAN dynamics, path integration process and cognitive map construction process. In this section, we give a description of these components in detail.

4.1 Local view cells

Local view cells are constructed as a vector, in which each element is associated with a cell unit in CAN. In visual processing module, if visual templates are sufficiently similar to recorded view templates, cognitive map model will inject an energy to associated cell unit. In contrast, significantly different images are treated as new templates and pushed into the vector. The inject energy will produce an activity packet at inactive zone in CAN model. This will help system to perform path integration and correct loop closure in

system. This change in cognitive map model caused from local view is described as

$$\Delta P_{x_i y_i \theta_i} = \sum_i \beta_{x_i y_i \theta_i} V_i \quad (1)$$

where β is a connection matrix from local view cells V_i to associated CAN cells P .

4.2 CAN dynamics

The core of the cognitive map model is a continuous attractor neural network, known as pose cell model. This model is widely used to simulate brain “GPS”. In Rat-SLAM, pose cells are arranged in (x, y, θ) coordinates, (x, y) is corresponding to a location in pose cell planes, and θ represents robot head direction. In addition, the dynamics of CAN structure ensures the activity P in pose cells to remain steady by following phases: an excitatory update, an inhibition and a normalisation.

4.2.1 Excitatory update

In this phase, a three-dimensional Gaussian distribution is employed to create an excitatory weight matrix ε , which drives activity P from each cell to all other cells in the pose cells matrix. The distribution is defined by

$$\varepsilon_{abc} = e^{-\frac{a^2+b^2}{\sigma_{xy}}} e^{-\frac{c^2}{\sigma_\theta}} \quad (2)$$

where σ_{xy} and σ_θ are variances for the distribution in (x, y) plane and θ dimension. Indexes a, b and c mean x, y, θ are within the distribution in pose cells co-ordinates. Then, the change of activity $\Delta P_{x,y,\theta}$ in pose cells is given by

$$\Delta P_{x,y,\theta} = \sum_{x_i=0}^{DimX} \sum_{y_i=0}^{DimY} \sum_{\theta_i=0}^{Dim\theta} P_{x_i,y_i,\theta_i} \varepsilon_{(a-x_i)(b-y_i)(c-\theta_i)} \quad (3)$$

where $DimX, DimY$ and $Dim\theta$ are three dimensions of pose cells, we set $DimX = DimY = 60$ and $Dim\theta = 36$ in our experiment.

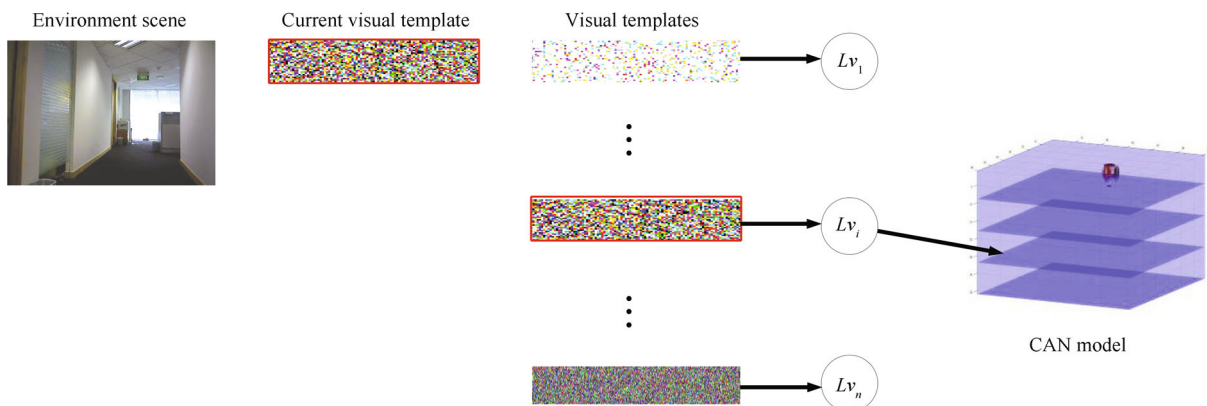


Fig. 4 Templates matching process for local view cells. ORB features are extracted from environment scenes as visual template which is compared against all visual templates associated with local view cells Lv_i . When current visual template matches a prior template in visual templates, the associated local view Lv_i fires and injects energy to cognitive map model. Otherwise, add a new local view cell Lv_j to Lv .

4.2.2 Inhibition

Each cell also restrains neighboring cells via an inhibitory weight matrix ξ . ξ has the same form as excitatory weight matrix, but non-positive weights. After excitatory phase, the opposite inhibition will result in convergence in network. By performing inhibition and adding a global inhibition μ , the change of activity is given by

$$\Delta P_{x,y,\theta} = \sum_{x_i=0}^{DimX} \sum_{y_i=0}^{DimY} \sum_{\theta_i=0}^{Dim\theta} P_{x_i,y_i,\theta_i} \xi - \mu \quad (4)$$

where ξ is inhibitory weight matrix and activity P in pose cell model is limited to non-negative value.

4.2.3 Normalization

Finally, the normalization will ensure the number of activity P in pose cells to remain at one. The activity after normalization is

$$P_{x,y,\theta}^{t+1} = \frac{P_{x,y,\theta}^t}{\sum_{x_i=0}^{DimX} \sum_{y_i=0}^{DimY} \sum_{\theta_i=0}^{Dim\theta} P_{x_i,y_i,\theta_i}^t}. \quad (5)$$

4.3 Path integration

The path integration shifts the pose cell activity away from the existing one based on raw odometry information. This process eliminated the need of tuning parameters in pose cell model. Furthermore, path integration process is independent of volatile robot velocity and sensory information. The activity update process is defined as

$$P_{x_i y_i \theta_i}^{t+1} = \sum_{a=\rho_x}^{\rho_x+1} \sum_{b=\rho_y}^{\rho_y+1} \sum_{\theta=\rho_\theta}^{\rho_\theta+1} \alpha_{abc} P_{(x_i+a)(y_i+b)(\theta_i+c)}^t \quad (6)$$

where ρ_x , ρ_y and ρ_θ are integer offset compared to current activity position (x_i, y_i, θ_i) . The activity $P_{x_i y_i \theta_i}^{t+1}$ depends on previous time step activity $P_{x_i y_i \theta_i}^t$ and a residue component α . The residue component is based on the fractional part of the offset. The integer offset is calculated by

$$\begin{bmatrix} \rho_x \\ \rho_y \\ \rho_\theta \end{bmatrix} = \begin{bmatrix} \lfloor \lambda_x v \cos \theta_i \rfloor \\ \lfloor \lambda_y v \sin \theta_i \rfloor \\ \lfloor \lambda_\theta \omega \rfloor \end{bmatrix} \quad (7)$$

where v is translational velocity, ω represents rotation velocity and θ_i is the preferred cell orientation. λ_x , λ_y and λ_θ are path integration constants.

4.4 Cognitive map construction

Cognitive map is a topological map that represents spatial relationship among landmarks of an environment. Each point in cognitive map can be represented by a triple

$$e_i = \{P_i, V_i, \mathbf{p}_i\} \quad (8)$$

where e_i , called experience, is a point in cognitive map. Pose cells P_i and local view cells V_i are associated to e_i . Finally, \mathbf{p}_i is the location of e_i within map space. In this

paper, if the distance between the pose cells of existing experiences and current pose cell reaches a threshold, then a new experience is created:

$$e_j = \{P_j, V_j, \mathbf{p}_i + \Delta p_{ij}\} \quad (9)$$

where the translation Δp_{ij} between two experiences is obtained from mobile base. Finally, when cognitive map model detects a loop closure in experience e_i , the change in the position of experience e_i is obtained by

$$\Delta p_i = \alpha \left[\sum_{j=1}^{N_f} (p_j - p_i - \Delta p_{ij}) + \sum_{k=1}^{N_t} (p_k - p_i - \Delta p_{ki}) \right] \quad (10)$$

where $\alpha = 0.5$ is a correction rate constant, N_f is the number of links from experience e_i to other experiences, N_t is the number of links from other experiences to experience e_i .

5 Experiment result

We evaluate the different aspects of our proposed algorithm in the following sections. In Section 5.1, we compare ORB algorithm with SIFT and the effectiveness of adopting RGB-D information. Next, we present the mapping result of our method in Section 5.2. In Section 5.3, we evaluate the performance of localization ability when facing noise data. The experiment data^[20] comes from an office in Singapore Research Institute, which contains many corridors, uniform furniture and moving people.

5.1 Feature extraction

In this section, we demonstrate ORB performance by making a comparison between SIFT and Tian's method^[20]. SIFT is one of the most popular feature extraction methods and a good fit for visual SLAM. Tian's visual processing method adopted RGB-D information to build a cognitive map successfully. To better illustrate the ability of ORB, SIFT and Tian's method, we select some events from experiment.

We firstly show the main idea of visual processing method of Tian's work based on RGB-D information in Fig. 5. From Fig. 5, intensity profiles of neighboring environment scenes are firstly extracted. Then, these profiles are processed by a sum of absolute differences (SAD) method to get the distance d_i . This distance is the sum of absolute differences between pixels value in these intensity profiles. Each distance d_i from both RGB and depth frames are finally assigned by using different weights to construct the distance d . It extracts one dimensional intensity profiles from both RGB and depth images. And these one dimensional intensity profiles are processed to calculate the distance between current image and recorded visual templates in local view cells. This method discards much information from physical environment, although it is simple and efficient. Unlike the above method, SIFT and ORB provided a better way

to extract features from incoming RGB frame to describe the surrounding environment.

In Fig. 6, image features based on SIFT and ORB are presented with circles and matched features are connected with lines. In the first row, the extracted features of neighboring frames are shown from both SIFT and ORB methods. It is obvious that these two methods provided the similar

image matching capability. In the second row, we specially selected a loop-closure circumstance from the experiment, the result shows that both methods can verify that these two images captured in different time representing the same location in the map. Different from SIFT, ORB is more suitable to extract features in objects that are at a middle and farther distance.

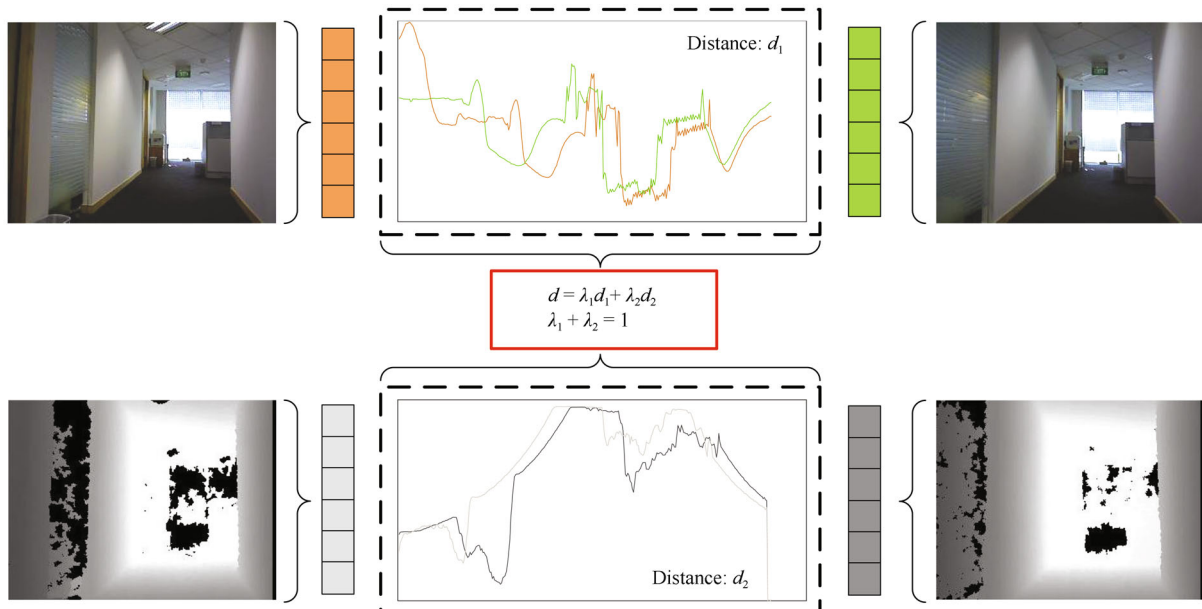


Fig. 5 A pair of neighboring RGB-D frames is showed. The top row is RGB information. The bottom row is depth information. Both RGB and depth information are captured simultaneously. And then, intensity profiles of neighboring environment scenes are extracted. These intensity profiles are processed by a sum of absolute difference to get a distance d_i . Distances d_i from RGB and depth images are weighted and then contributed to d , which is used to distinguish different scenes.



Fig. 6 Example of features matched by using SIFT (pair on the left) and ORB descriptors (pair on the right) in the first and second rows. The first row presents neighboring frames captured by camera. The second row presents a loop closure event in experiment. It is obvious that both SIFT and ORB can provide sufficient corresponding matched points.

Furthermore, one of the advantages of ORB method is its higher efficiency on a standard CPU, this can be compared with SIFT which needs a lot of calculation time. In our experiment, the feature extraction process was executed in a single thread running on an Intel i7 3.6 GHz processor. The parameters of ORB implementation are to extract 500 key points, at 8 different pyramid levels with a scale factor of 1.2.

We compare the processing time for both SIFT and ORB by testing them on our experiment data set including 3 732 frames. The comparison results are summarized in Table 1.

Our results show that ORB features are almost as reliable as SIFT features for detecting loop closure and matching neighboring frames. As advantages, ORB features not only are invariant to image scale and rotation, but also are much faster to process, speeding up the visual processing method.

Table 1 Comparison of visual processing time

Methods	ORB	Tian ^[20]	SIFT
Visual information	RGB	RGB-D	RGB
Time per frame (ms)	4.77	4.54	196.83
Total time for 3 732 frames (s)	16.43	14.23	734.46

5.2 Cognitive map

In this section, cognitive mapping results are showed to demonstrate the performance of our proposed method. As shown in Fig. 7, the mapping process includes three major steps: 1) The external RGB images and internal self-motion are used to generate and shift active packets in the pose cell model. 2) Local view cells correct accumulative errors caused by the path integration when a loop closure is formed. 3) The cognitive map will persistently update throughout the process. In the following, we will describe these processes in the local view cells, pose cells and cognitive map construction.

5.2.1 Local view cell activity

The learned view template versus frame generated from RatSLAM using one-dimensional profiles and ORB features are showed in this section.

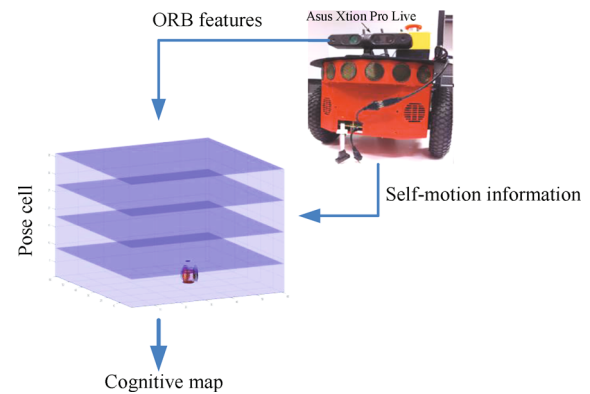


Fig. 7 Cognitive mapping process

The local view cell learned 1 365 visual templates during mapping process, as shown in Fig. 8 (a). The graph shows the learned visual template versus frame generated from ORB based visual processing module. The y -axis indicates the addition of learned visual templates and x -axis indicates the number of incoming frames captured by camera. Duration of without new template means that either the robot travels along places that have been learned, or the robot stays static. Furthermore, the number of newly added local view cells will gradually decrease after robot loops the environment several times. Compared with Fig. 8 (b), which is generated from one dimensional intensity profiles^[19], it is obvious that many false positives have been produced, directly leading to wrong loop closure and failures in spatial representation. Hence, it is desirable to use ORB to construct a more accurate map.

5.2.2 Pose cell activity

During mapping process, there is an active packet in pose cell model. The active packet represents robot's location. Normally, it moves around according to self-motion information from mobile base. However, there are two typical events in the pose cell model during mapping process, as shown in Fig. 9.

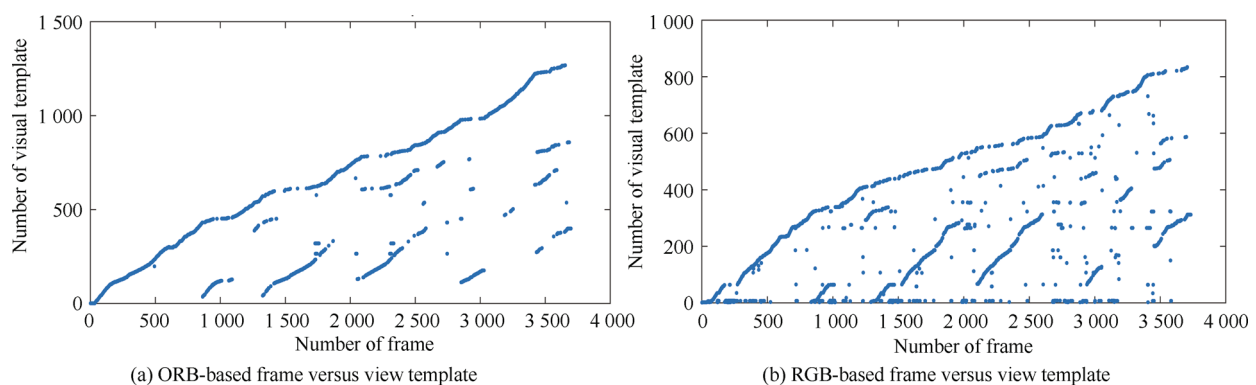


Fig. 8 Activity of local view cells during mapping process. Graphs (a) and (b) are view template versus frame generated from ORB features and intensity profiles, respectively. While making a comparison with (a) and (b), it is obvious that ORB can help RatSLAM significantly reduce the number of false positives.

Generally, the number of cell units in pose cell model is inadequate to present all corresponding places in physical environment. There is a one-to-many relationship: One cell unit can have many physical environment places associated to it. Therefore, pose cells are arranged with a recurrent structure to keep continuity of pose cell model. Hence, sometimes partially activity packet makes a leap to opposite face in the pose cell, as shown in Fig. 9(a).

Local view cells play an important role in loop closure

process. When robot enters a place that has been reached before, active local view cells transmit an energy to associated pose cell. Fig. 9(b) and Fig. 9(c) show that system encounters a familiar scene, and then local view cells inject energy to pose cells, eventually, create two peaks in pose cells. However, the pose cell model is capable of maintaining or correcting its beliefs, the dynamics of pose cells filters ambiguity caused by the similar visual cues and converges pose cell model into one single peak.

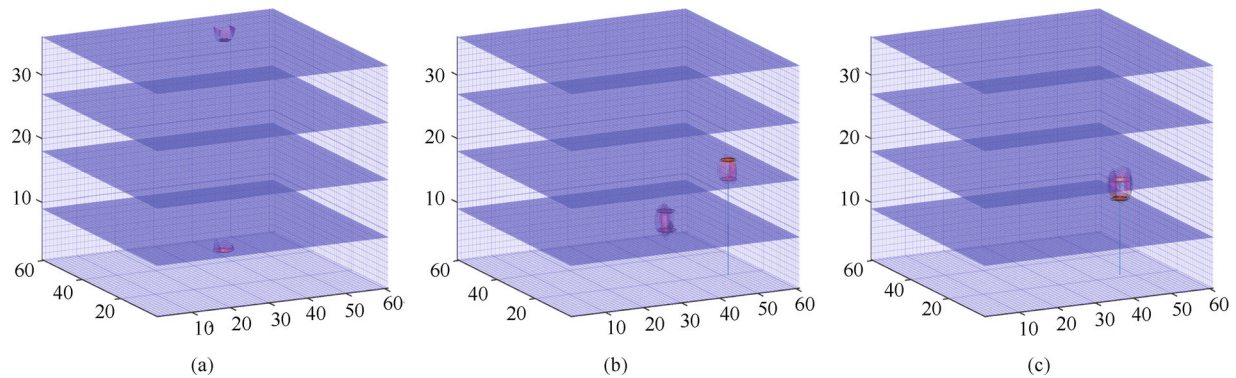


Fig. 9 Activity packets are present by the orange zone, the lighter part means higher activation level. In (a), one active packet shifts to opposite face to keep the consistency active packet. In (b), when the robot recognizes a sufficient similar environment scene, local view cells inject energy into pose cells and then two active packets emerge in pose cells. In (c), only one packet has survived after both two activity packets undergo CAN internal dynamics.

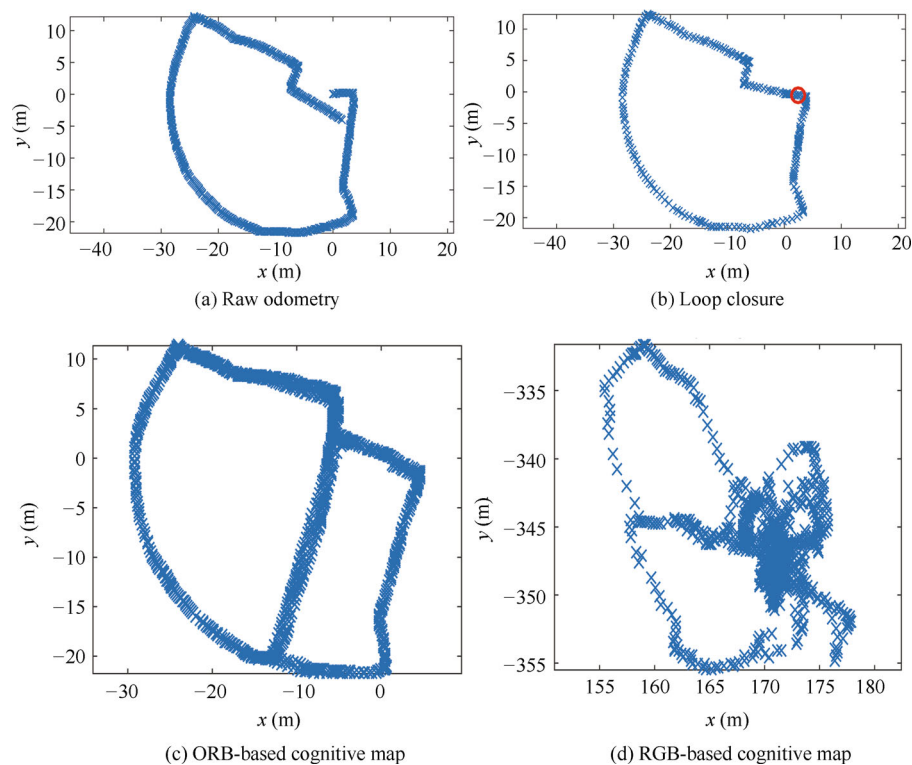


Fig. 10 (a) Self-motion information directly obtained from mobile base encoders contains “drift” phenomenon. (b) System detects loop closure and correct the error. The red circle means the robot current place. (c) and (d) are cognitive maps generated from mapping system based on ORB features and one-dimensional intensity profile respectively.

5.2.3 Cognitive map construction

The mapping system does not track visual features across several frames and does not take geometric relationship of landmarks into consideration. It only connects and records the simple spatial relationship of places in environment.

However, the map will constantly do map correction during the whole process. When the robot reaches a familiar environment scene, active local view cell injects energy to associated pose cell, causing the robot to localise its current pose. These behaviors prevent the robot from losing its way, as shown in Figs. 10 (a) and 10 (b). In Fig. 10 (a), it is obvious that robot falsely locates itself when it returns to previously explored area. This is because self-motion information from mobile base contains errors, directly leading to the “drift” phenomenon. But, RatSLAM can detect loop closure and reset its position, after local view cell stimulates associated pose cell in Fig. 10 (b).

For comparison purpose, a map is built by one dimensional intensity profiles^[19], as shown in Fig. 10 (d). It has been observed that this method failed to perform loop closure and is unable to construct a correct map of office environment. And conversely, in Fig. 10 (c), RatSLAM can

eliminate the ambiguous scenes and correct loop closure, eventually, construct a correct cognitive map with the help of ORB features.

5.3 Global localization

In the above section, we have studied the effectiveness of different visual processing methods for cognitive map building. In this section, how different visual processing methods affect localization results will be focused on. In order to test the ability of localization, we randomly selected 1244 frames from the raw database as the test data set. From Table 1 in Section 5.1, SIFT algorithm needs to spend about 200 ms extracting features even for each frame. This slow calculation makes it hard to fit real-time robotics applications. So we only compare the localization results based on ORB and Tian’s method^[20] in the following experiment. The experiment results include two cases:

- 1) Images are clean without any noise.
- 2) Images are simulated to rotate with small degrees.

For the first case, the comparison results are summarized in Table 2.



Fig. 11 This figure shows a snapshot of environment. (a) Original image selected from experiment. (b), (c) and (d) are environment scenes and are clockwise rotated by 3, 5 and 7 degrees, respectively. With the increase of rotation angles, it is more difficult for system to locate itself within the map.

From Table 2, it is observed that a satisfactory accuracy of localization based on ORB method can be achieved which only requires RGB information of images (about 99.6%). This accuracy is almost the same as Tian's method in which both RGB and depth information are used together. It is concluded that ORB is almost as reliable as adopting RGB-D information. These results also demonstrate that ORB features are capable of distinguishing various scenes in environment.

For the second case, in order to study the robustness of ORB features, we first simulate images with noise by rotating the clean images with a few small orientations as shown in Fig. 11.

The localization performance for the simulated noise is given in Table 3. Table 3 shows that both ORB and Tian's method can provide the good localization performance with the accuracy above 95% when images are rotated by an angle up to 7 degrees. This implies that ORB features are capable of dealing with perceptual ambiguity in office environment without depth information.

Table 2 Comparison of localization result

Method	Visual information	Trials	Miss matched images	Accuracy (%)
ORB	RGB	1 244	6	99.5
Tian ^[20]	RGB-D	1 244	4	99.7

Table 3 Localization result (with rotation noise)

Method	Visual information	Rotation angles	Miss matched images	Accuracy (%)
ORB	RGB	1°	7	99.4
		2°	9	99.3
		3°	13	99.0
		4°	24	98.0
		5°	35	97.4
		7°	52	95.8
Tian ^[20]	RGB-D	1°	4	99.7
		2°	4	99.7
		3°	4	99.7
		4°	4	99.7
		5°	4	99.7
		7°	16	98.7

6 Conclusions

In this paper, we applied ORB feature extraction approach into RatSLAM system to build a cognitive map for a mobile robot. In order to test ORB method, the performance of extracted features has been demonstrated by comparing a few different visual processing approaches. We also compared the performance of different visual processing methods such as SIFT approach and existing feature extraction methods^[19, 20] for both RGB and RGB-D signals. The feature extraction section showed that ORB is not only suitable for matching neighboring frames with in-

plane camera motion, but also satisfying real-time visual processing requirement. In global localization section, the proposed brain-inspired SLAM system is able to realize the localization in the map with the help of ORB features. Furthermore, the cognitive mapping results verified that ORB can significantly enhance the robustness of SLAM system in indoor environment. Specifically, the proposed SLAM system can greatly reduce false positives and repeatedly correct loop closure even facing accumulative odometry error, and eventually, constructs a cognitive map.

Open Access

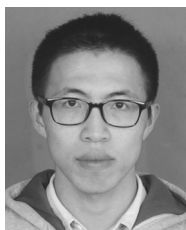
This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- [1] E. C. Tolman. Cognitive maps in rats and men. *Psychological Review*, vol. 55, no. 4, pp. 189–208, 1948.
- [2] J. O'Keefe, J. Dostrovsky. The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely-moving rat. *Brain Research*, vol. 34, no. 1, pp. 171–175, 1971.
- [3] M. Fyhn, S. Molden, M. P. Witter, E. I. Moser, M. B. Moser. Spatial representation in the entorhinal cortex. *Science*, vol. 305, no. 5688, pp. 1258–1264, 2004.
- [4] T. Hafting, M. Fyhn, S. Molden, M. B. Moser, E. I. Moser. Microstructure of a spatial map in the entorhinal cortex. *Nature*, vol. 436, no. 7052, pp. 801–806, 2005.
- [5] B. L. McNaughton, F. P. Battaglia, O. Jensen, E. I. Moser, M. B. Moser. Path integration and the neural basis of the cognitive map. *Nature Reviews Neuroscience*, vol. 7, no. 8, pp. 663–678, 2006.
- [6] M. C. Fuhs, D. S. Touretzky. A spin glass model of path integration in rat medial entorhinal cortex. *Journal of Neuroscience*, vol. 26, no. 16, pp. 4266–4276, 2006.
- [7] P. Lavenex, D. G. Amaral. Hippocampal-neocortical interaction: A hierarchy of associativity. *Hippocampus*, vol. 10, pp. 420–430, 2000.
- [8] F. Savelli, J. J. Knierim. Hebbian analysis of the transformation of medial entorhinal grid-cell inputs to hippocampal place fields. *Journal of Neurophysiology*, vol. 103, no. 6, pp. 3167–3183, 2010.
- [9] T. Solstad, E. I. Moser, G. T. Einevoll. From grid cells to place cells: A mathematical model. *Hippocampus*, vol. 16, no. 12, pp. 1026–1031, 2006.

- [10] J. Hu, H. J. Tang, K. C. Tan, H. Z. Li. How the brain formulates memory: A spatio-temporal model research frontier. *IEEE Computational Intelligence Magazine*, vol. 11, no. 2, pp. 56–68, 2016.
- [11] E. T. Rolls, S. M. Stringer, T. Elliot. Entorhinal cortex grid cells can map to hippocampal place cells by competitive learning. *Network: Computation in Neural Systems*, vol. 17, no. 4, pp. 447–465, 2006.
- [12] E. I. Moser, E. Kropff, M. B. Moser. Place cells, grid cells, and the brain's spatial representation system. *Annual Review of Neuroscience*, vol. 31, pp. 69–89, 2008.
- [13] M. L. Yuan, B. Tian, V. A. Shim, H. J. Tang, H. Z. Li. An entorhinal-hippocampal model for simultaneous cognitive map building. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, AAAI, Austin, USA, pp. 586–592, 2015.
- [14] N. Cuperlier, M. Quoy, P. Gaussier. Neurobiologically inspired mobile robot navigation and planning. *Frontiers in Neurobotics*, vol. 1, pp. 3, 2007.
- [15] A. Jauffret, N. Cuperlier, P. Gaussier, P. Tarroux. Multimodal integration of visual place cells and grid cells for navigation tasks of a real robot. In *Proceedings of the 12th International Conference on Simulation of Adaptive Behavior*, Springer, Odense, Denmark, pp. 136–145, 2012.
- [16] W. W. Huang, H. J. Tang, B. Tian. Vision enhanced neuro-cognitive structure for robotic spatial cognition. *Neurocomputing*, vol. 129, pp. 49–58, 2014.
- [17] W. Hess, D. Kohler, H. Rapp, D. Andor. Real-time loop closure in 2D LIDAR SLAM. In *Proceedings of IEEE International Conference on Robotics and Automation*, IEEE, Stockholm, Sweden, pp. 1271–1278, 2016.
- [18] G. Grisetti, C. Stachniss, W. Burgard. Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 34–46, 2007.
- [19] M. J. Milford, G. F. Wyeth. Mapping a suburb with a single camera using a biologically inspired SLAM system. *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1038–1053, 2008.
- [20] B. Tian, V. A. Shim, M. L. Yuan, C. Srinivasan, H. J. Tang, H. Z. Li. RGB-D based cognitive map building and navigation. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, Tokyo, Japan, pp. 1562–1567, 2013.
- [21] V. A. Shim, B. Tian, M. L. Yuan, H. J. Tang, H. Z. Li. Direction-driven navigation using cognitive map for mobile robots. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, Chicago, USA, pp. 2639–2646, 2014.
- [22] J. Engel, T. Schöps, D. Cremers. LSD-SLAM: Large-scale direct monocular SLAM. In *Proceedings of the 13th European Conference on Computer Vision*, Springer, Zurich, Switzerland, pp. 834–849, 2014.
- [23] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, W. Burgard. An evaluation of the RGB-D SLAM system. In *Proceedings of IEEE International Conference on Robotics and Automation*, IEEE, Saint Paul, USA, pp. 1691–1696, 2012.
- [24] K. Konolige, M. Agrawal. FrameSLAM: From bundle adjustment to real-time visual mapping. *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1066–1077, 2008.
- [25] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [26] H. Bay, T. Tuytelaars, L. Van Gool. SURF: Speeded up robust features. In *Proceedings of the 9th European Conference on Computer Vision*, Springer, Graz, Austria, pp. 404–417, 2006.
- [27] S. N. Sinha, J. M. Frahm, M. Pollefeys, Y. Genc. GPU-based video feature tracking and matching. In *Proceeding of Workshop on Edge Computing Using New Commodity Architectures*, Chapel Hill, USA, 2006.
- [28] E. Rosten, T. Drummond. Machine learning for high-speed corner detection. In *Proceedings of the 9th European Conference on Computer Vision*, Springer, Graz, Austria, pp. 430–443, 2006.
- [29] E. Rosten, R. Porter, T. Drummond. Faster and better: A machine learning approach to corner detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 105–119, 2010.
- [30] M. Calonder, V. Lepetit, C. Strecha, P. Fua. Brief: Binary robust independent elementary features. In *Proceedings of the 11th European Conference on Computer Vision*, Springer, Crete, Greece, pp. 778–792, 2010.
- [31] S. K. Vipparthi, S. K. Nagar. Local extreme complete trio pattern for multimedia image retrieval system. *International Journal of Automation and Computing*, vol. 13, no. 5, pp. 457–467, 2016.
- [32] Z. H. Wang, Q. F. Song, H. M. Liu, Z. Q. Huo. Absence importance and its application to feature detection and matching. *International Journal of Automation and Computing*, vol. 13, no. 5, pp. 480–490, 2016.
- [33] G. Klein, D. Murray. Parallel tracking and mapping for small AR workspaces. In *Proceedings of the 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, IEEE, Nara, Japan, pp. 225–234, 2007.
- [34] D. Gálvez-López, J. D. Tardos. Bags of binary words for fast place recognition in image sequences. *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.
- [35] E. Rublee, V. Rabaud, K. Konolige, G. Bradski. ORB: An efficient alternative to SIFT or SURF. In *Proceedings of IEEE International Conference on Computer Vision*, IEEE, Barcelona, Spain, pp. 2564–2571, 2011.
- [36] C. Harris, M. Stephens. A combined corner and edge detector. In *Proceedings of the 4th Alvey Vision Conference*, Manchester, UK, pp. 147–151, 1988.

- [37] R. Mur-Artal, J. M. M. Montiel, J. D. Tardós. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.
- [38] R. Mur-artal, J. D. Tardós. Fast relocalisation and loop closing in keyframe-based SLAM. In *Proceedings of IEEE International Conference on Robotics and Automation*, IEEE, Hong Kong, China, pp. 846–853, 2014.

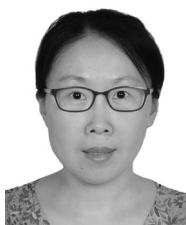


Sun-Chun Zhou received the B.Sc. degree from China University of Mining and Technology, China in 2015. He is currently a master student in the College of Computer Science, Sichuan University, China.

His research interests include cognitive robots, SLAM and robotic navigation.

E-mail: zhousunchun@gmail.com

ORCID ID:0000-0002-5826-2105



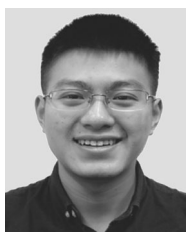
Rui Yan received the B.Sc. and M.Sc. degrees from Department of Mathematics, Sichuan University, China in 1998 and 2001, respectively, and received the Ph.D. degree from Department of Electrical and Computer Engineering, National University of Singapore, Singapore in 2006. She was a postdoctoral research fellow in the University of Queensland, Australia, from

2006 to 2008, and a research scientist with the Institute for Infocomm Research, A*STAR from 2009 to 2014. Now she is a professor in the College of Computer Science, Sichuan University, China.

Her research interests include intelligent robots, neural computation, nonlinear control and complex systems analysis.

E-mail: ryan@scu.edu.cn (Corresponding author)

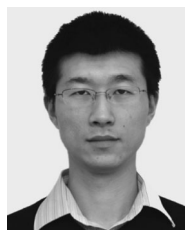
ORCID ID: 0000-0003-0048-3092



Jia-Xin Li received the B.Sc. degree from Sichuan University, China in 2016. He is currently a Ph.D. degree candidate in the College of Computer Science, Sichuan University, China.

His research interests include SLAM, robotic navigation and cognitive robots.

E-mail: ljxin1993@gmail.com



Ying-Ke Chen received the Ph.D. degree from Aalborg University, DK in 2013. He was a post-doctoral research associate in Queen's University Belfast, USA and Georgia University, US, respectively, before joining Sichuan University. He is a lecturer at College of Computer Science, Sichuan University, China.

His research interests include intelligent agents, decision making and their applications in autonomous systems.

E-mail: chen.yk@scu.edu.cn



Huajin Tang received the B.Eng. degree from Zhejiang University, China in 1998, received the M.Eng. degree from Shanghai Jiao Tong University, China in 2001, and received the Ph.D. degree from the National University of Singapore, Singapore in 2005. He was a system engineer with STMicroelectronics, Singapore, from 2004 to 2006. From 2006 to 2008, he was

a post-doctoral fellow with the Queensland Brain Institute, University of Queensland, Australia. Since 2008, he was head of the Cognitive Computing Group and the Robotic Cognition Lab, Institute for Infocomm Research, A*STAR, Singapore. He is currently a National Youth-1000 Talent Distinguished professor and the director of the Neuromorphic Computing Research Center, Sichuan University, China. He has co-authored two monographs by Springer and over 70 international journal and conference papers. His research work on Brain GPS has been reported by *MIT Technology Review* on 2015. He has received the 2011 Institute for Infocomm Research Role Model Award, 2016 IEEE CIS Outstanding TNNLS Paper Award. He was the Program Chair of the 6th and 7th IEEE CIS-RAM, and the Co-Chair of 2016 International Workshop on Neuromorphic Computing and Cyborg Intelligence. He is currently an associate editor of *IEEE Transactions on Neural Networks and Learning Systems*, *IEEE Transactions on Cognitive and Developmental Systems* and *Frontiers in Neuromorphic Engineering*.

His research interests include neuromorphic cognitive computing, neuromorphic hardware and neuro-cognitive robotics.

E-mail: htang@scu.edu.cn