An Approach to Localization and Mapping Based on Centers of Box-detectors

Li LUO¹, Zhen LIU¹, Qi WEI², Fei QIAO²*, Xin-jun LIU³, Gui-jin WANG² and Hua-zhong YANG²

¹School of Electronic Information Engineering, Beijing Jiaotong University, China
²Dept. of Electronic Engineering, Tsinghua University, China
³Dept. of Mechanical Engineering, Tsinghua University, China
*Corresponding author

Keywords: Semantic mapping, Object detection, SLAM, CNN.

Abstract. For the next generation of intelligent robots to communicate with human and surroundings, it is necessary to contain both structures and semantics in the map. However, ignoring the relevance between structures and semantics, the substantial part of researches to address this problem focus on SLAM (Simultaneous Localization and Mapping) and object detection in parallel. In this work, a novel approach is presented to fusing object detection and mapping together, which takes advantage of object detection based on CNN (Convolution Neural Network) and maps by utilizing these detected objects. The system is capable of capturing indoor comprehensive instances using a RGB-D camera and consistently building point cloud map based on these known objects' centers. The feasibility and scalability of the system are demonstrated through experiments in a real-world scene with abundant recognizable objects.

Introduction

Surroundings geometry information is necessary for traditional indoor-robots to localize themselves and avoid obstacles when moving in the unknown environment. It can operate fluidly with the support of VSLAM (Visual Simultanous Localization and Mapping) that has made impressed progress in recent years [1]. Some works [2, 3] even can be applied in mobile terminals in real-time. Nevertheless, these researches in visual mapping neglect the rich semantic information contained in surroundings, which is vital component of maps in the future indoor robot applications.

The aim of this paper is to combine object detection and RGBD SLAM together, with a novel solution based on the high accuracy of neural network detection and the SLAM to keep the recognized objects constantly located in the scene. Furthermore, the classic SLAM is not the final target, it is the significance that combing semantics and SLAM deeply to make that SLAM helps semantics and semantics helps SLAM. To achieve this, a paradigm for combining object detection and mapping without extracting any kind of low-level hand-craft features (such as SIFT, SURF or ORB) is proposed in this work. The approach to use the object predictions from the continuous RGB-D camera movements to extract the center-points when the amount of recognized objects is up to or beyond four in a single frame that can be utilized to localize the camera pose. The CNN to detect the objects in the experiments is the SSD(Single-Shot Multi-box Detector)[4] network because the accuracy of object detection and running time have excellent performance on public datasets such as VOC2012[11].

As shown in Fig. 1, the system generates a point cloud map of the surrounding environment that has enough recognized objects. The significant distinction to other work is that the system jumps over the feature points calculation and description and produces the incrementally built map directly with the recognized objects. Moreover, due to the object models containing both geometry and semantic information in the final map do not require precise surface structure messages, the system needn’t spend hardware resources storing object models. Meanwhile, the resulting map of the system could be
explored for more applications such as navigation, human-computer interaction and scene understanding. This approach does not require a priori known 3D model of every recognized object like [12]. Besides, the system does not need extra GPU resources except CNN-based object detection training.

This approach can simultaneously detect objects and localize. It differs from these works[5,6] which build semantic maps at last depending on existing SLAM and these works[6,7,8] which enhance recognition performance in a consistent environment by leveraging SLAM or combining SLAM and object recognition together to improve the robustness of the whole system.

This research makes the following contributions:
(i) the capability to propose boxes containing object candidates based on the relevancy among frames is presented. Leveraging the centers of box-detector, the system localizes the camera and maps directly without any hand-craft feature detectors and descriptors.
(ii) Several results validating the improved of object detection performance are presented: The system is compared against CNN’s object detection.

In the rest of this paper, the system will be first illustrated in Method part. The experiments and discussion of results will be given in Experiment part.

Method

The proposed system is able to recognize several objects in the scene, aggregating box-detectors to localize the camera and map. The raw images are sent to detect objects and then the detected messages are utilized to localize camera and map with the every depth frames. The output of the system is the point cloud map of surrounding environment.

The system is overviewed in Figure. 1 and detailed below. 1) The current RGB frame captured by the kinect2 camera is fed into the SSD network. 2) Then the system tracks the objects’ trajectory in the adjoining frames. 3) The frame containing N object labels and the current depth frame are sent into the module of extracting the center of objects. 4) The M objects contained in the last frame are matched with the N objects contained in the current frame. If the successfully matched number n is greater than 4, the problem that calculates the rotate and translation matrixes can be regarded as the PnP (perspective n points) problem. And finally the current point cloud can do registration with the existed map.

Object Detection for the System

In the Localization module, the ability of detecting objects in the frames decides whether the rotation and translation matrixes in the frames can be calculated. Therefore, every individual object which is recognized in the object detection module is crucial in the system. Which means, the high
accuracy and stability of object detection should be guaranteed. As far as we know, the
proposal-based CNN approaches have shown excellent performance in object recognition.

The system adopts the Single-shot Multi-box Detector network architecture proposed by Wei Liu [4]. SSD’s framework is implemented in Caffe [9] and the architecture is based on VGG-16 network [10], but with the addition of convolutional feature layers which are trained for predicting detections in different scales. SSD outputs a class label and a confidence probability for every bounding box fitted well with the object. The network is trained on the PASCAL VOC dataset which has 20 object classes and another background class.

In order to improve the stability of the system, pre-processing is used to do with the outputs of SSD. It takes the advantage of priori knowledges that the objects in the same class share almost the same size. And the specific method is shown as follows.

Each recognized object \( O_i \in l \) in the frame \( F_t \) is supposed to have a default size that is composed of two parameters, \( H_i \) and \( W_i \). \( O_i \) is the \( i \)th object in the frame \( F_t \) captured by the camera at time \( t \). \( l \in L \), which is one of object classes in the PASCAL VOC dataset. And \( H_i \) and \( W_i \) represent respectively the height and the width of the object in class.

The depth \( \text{depth}_i \) of the centre point in the detected rectangle are calculated to represent the average depth of \( O_i \). As shown in Fig 2, the supposed size \( h_i \) and \( w_i \) are shown as fellows,

\[
h_i = \frac{f}{\text{depth}_i} \cdot H_i
\]

\[
w_i = \frac{f}{\text{depth}_i} \cdot W_i
\]

Then, if \( A_i \times h_i \times w_i < \delta \), the detected rectangle will be replaced with the predicted rectangle which is the output of the object tracking module. \( A_i \) represents the detected rectangle area, \( \delta \) represents threshold that is pre-defined.

![Figure 2. Improved Object Detection.](image)

Considering the continuity of the indoor-robot movements, it should have strong relevancy among the captured pictures, especially the adjoining frames. It is to say, the object detected in the last frame wouldn’t suddenly disappear in the current frame without a warrant. Actually, the results of SSD network exactly show the instability when the network operates in a real-world scene.

Assuming the left-top coordinate of the rectangle containing object \( O_i \) in frame \( F_t \) is \( (x_{j_0}, y_{j_0}) \), the localization of \( O_i \) in frame \( F_{t+1} \) is predicted.
In formula (3) and (4), \( (x_{f(t+1)}, y_{f(t+1)}) \) represents the predicted left-top coordinate of the rectangle containing object \( O_i \) in the frame \( F_t \). As the camera moves slowly and uninterruptedly in the indoor room, \( (x_{f(t+1)}, y_{f(t+1)}) \) is utilized to simulate the objects’ movement among frames. So the four vertexes of the predicted rectangle can be got in the next frame.

After getting the recognized objects between the sequential frames, Calculating the rotation and translation matrixes is actually the PnP problem because the depth of object’s center is known in the corresponding depth images. Due to the lack of pose optimisation and loop closure, this part is called as Visual Odometry instead of SLAM. This work is a little bit different from semantic SLAM since that requires SLAM helping semantics and semantics helping SLAM. The system is demonstrated that it leverages object messages to locate the camera but can be extended to improve object detection precision by locating the objects.

The camera pose and the depth images are used to build a semantic map. For each arriving frame which has enough objects, Visual Odometry tracks the camera pose via the SolvePnP to yield the matrixes \( R \) and \( t \). New point clouds are added into the map using this camera pose.

**Experiment**

The SSD network is initialized with weights from Wei Liu [4] trained for object detection on the PASCAL VOC 2012 dataset [11]. And then the outputs are fed into object tracking module for optimization. The capabilities of object tracking is demonstrated in the real-world scene test. The scene contains 6 objects which can be recognized by SSD. The number of total frames is 33. In contrast to SSD, this experiment respectively counts the number of recognized objects processed by SSD-only and by SSD-Object Tracking. In case, the error-recognition is treated as a missed object.

In the Fig. 2, pictures (a), (b), (c) and (d) are the direct outputs of SSD network whose inputs are the sequential frames captured by Kinect2. In them, the different colours represent different classes, e.g. the light green rectangle in picture (a) contains a bottle while the red rectangle contains a TV monitor. In (c), the TV monitor cannot be recognized and even be regarded as chair at a probability of 0.73. Pictures (e), (f), (g) and (h) were the results that takes the post-processing considering the continuity of the indoor robot movements. The error-recognition and misrecognition problems were solved to a certain extent shown in picture (g) and picture (h).

The results are shown in Fig. 3. In the first 13 frames, the both methods perform fairly well since SSD-Object Tracking is limited by the performance of SSD in last frames. However, the superiority of SSD-Object Tracking is shown after the 18th frame in which SSD detects the all 6 objects while SSD-only method fluctuates violently. It is due to whether the relevance of adjacent frames is taken into consideration. The average accuracy of SSD-only in this sequence is 47.9% while the average accuracy of SSD-Object Tracking is 76.7%. The performance has been increased by about 28% in the experiment. The accuracy of object detection is evaluated by 28% after considering the relevance among frames.
The shortcoming of SSD-Object Tracking is reflected in the constraints of SSD as mentioned above. In more detail, SSD-Object Tracking is an improvement to SSD-only, and the ability of object detection is fundamentally depended on SSD. Nevertheless, SSD-Object Tracking still goes for the application scenarios where the camera moves consistently and slowly. This hypothesis provides SSD-Object Tracking with the premise and actually it is very common in localization with camera.

Considering the characteristics of the system, it requires that the scene is supposed to contain enough recognized objects. There is almost no suitable public dataset for the system verification. The system are tested in different kinds of scenes, as shown in Fig. 4. The point cloud map which is the registration of 5 frames shows that the system can build the map. On the other hand, another test are done to evaluate the robustness of the system. The experiment maps the same scene with different number of frames and the results (frames =5, 10, 20, 30) are shown in Fig. 5. That can come to a conclusion qualitatively that the precision is decreasing as the total number of frames is increasing. The reason is obvious that the errors in each frame accumulate constantly as the number of frames increases.

Figure 4. The results of localisation in real-world scene. The two images in each row are both from the same scene and the difference between them is the view angle.
Conclusion

A novel approach is presented for simultaneous object detection and localization. This approach confirms the expectation, which using the object detection and depth message to provide spatial and semantic information can localize directly instead of low-level features like pixels and feature points. The number of object classes that the system can recognize is up to 20.

However, it still remains problems which will be solved in the future work. The bottleneck of the system is the robustness and accuracy of object detection. It cannot handle a large scale scene for the system as to the lack of pose optimisation and loop closure. Therefore, the future work could incorporate breakthroughs in object detection, and add optimisation strategies in the system.

Reference

[1] Grisetti, G., Kummerle, R., Stachniss, C., & Burgard, W. (2011). “A Tutorial on Graph-Based SLAM,” IEEE Intelligent Transportation Systems Magazine, 2(4), pp. 31-43.

[2] Engel, J., Schöps, T., & Cremers, D. (2014). “LSD-SLAM: Large-Scale Direct Monocular SLAM,” ECCV, pp. 834-849.

[3] Engel, J., Sturm, J., & Cremers, D. (2013). “Semi-Dense Visual Odometry for a Monocular Camera,” ICCV, pp. 1449-1456.

[4] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. (2016). “SSD: Single Shot MultiBox Detector,” ECCV, pp. 21-37.

[5] McCormac, J., Handa, A., Davison, A., & Leutenegger, S. (2016). SemanticFusion: Dense 3D Semantic Mapping with Convolutional Neural Networks. [Online]. Available: http://arxiv.org/pdf/1609.05130.pdf

[6] Sunderhauf, N., Pham, T., Latif, Y., Milford, M., & Reid, I. (2016). “Meaningful Maps – Object-Oriented Semantic Mapping.” [Online]. Available: http://arxiv.org/pdf/1609.07849.pdf

[7] Pillai, S., & Leonard, J. J. (2015). “Monocular SLAM Supported Object Recognition,” Computer Science, 2015.
[8] Dorian Gálvez-López*, 1, Marta Salas 1, Juan D. Tardós, J.M.M. Montiel. “Real-time monocular object SLAM,” Robotics and Autonomous Systems, vol. 75. pp. 435-449. 2016.

[9] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding.” arXiv preprint arXiv:1408.5093, 2014.

[10] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” [Online]. Available: https://arxiv.org/abs/1409.1556/

[11] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes (VOC) challenge,” International Journal of Computer Vision, no. 2, pp. 303–338, 2010.

[12] R. F. Salas-Moreno, R. A. Newcombe, H. Strasdat, P. H. J. Kelly, and A. J. Davison, “SLAM++: Simultaneous Localisation and Mapping at the Level of Objects,” in CVPR, 2013, pp. 1352-1359.