Traffic Sign Recognition Algorithm for Car-like Robot Avrora Unior

Nikita Nikiforov[1], Ksenia Shabalina[1], Artur Sagitov[1], Kuo-Hsien Hsia[2], Evgeni Magid[1]
[1]Laboratory of Intelligent Robotic Systems (LIRS), Intelligent Robotics Department, Higher Institute of Information Technologies and Intelligent Systems, Kazan Federal University, Russia
[2]Department of Electrical Engineering, National Yunlin University of Science and Technology, Taiwan
E-mail: NANikiforov1337@gmail.com, ks.shabalina@it.kfu.ru, sagitov@it.kfu.ru, khhsia@yuntech.edu.tw, magid@it.kfu.ru
http://kpfu.ru/robolab.html

Abstract

Achieving high accuracy of traffic signs detection and recognition is difficult in real-time and is heavily influenced by non-ideal environment conditions. In this paper, we propose to combine a set of Haar cascades that had been trained on a large number of samples and could recognize different types of road signs in different positions and orientations. We use feature detection and feature matching in the process of traffic sign type identification. Our algorithm was validated on Avrora Unior robot model in a simulated environment within Gazebo.

Keywords: traffic sign recognition algorithm, Avrora Unior, car-like robot, Gazebo, simulation.

1. Introduction

Traffic sign recognition is an important task for unmanned cars and car-like robots[1]. Since road traffic is almost entirely formed by human drivers on a road, it is required for mobile unmanned ground vehicles (UGV) to strictly follow state traffic regulations and to guarantee road safety for other road users. Due to this, many computer vision tasks have become classic tasks of intelligent road agents development: traffic signs[2], traffic lights[3], and road markings recognition[4], determining speed and direction of movement of other road users[5].

In this paper we focus on traffic signs recognition task for car-like UGV Avrora Unior[6]. This is necessary in order to enable autonomous path planning[7] and locomotion possibilities of the robot within public roads[8] as well as applying parking[9] and overtaking[10] algorithms.

The rest of the paper is organized as follows: Section 2 describes the task and cascade training. Section 3 explains operation of the algorithm and its components. In Section 4 we discuss experimental results. Finally, we conclude in Section 5.

2. Traffic Sign Recognition Problem

To tackle the problem of traffic sign recognition, we have constructed shape detection cascading classifier. As cascading classifiers need to be trained with several hundred positive detection examples and several negative examples (for every particular sign type) we needed a decent training set. We have selected GTRSB[11] database as it contains convenient image format, images have small sizes and additional support information of objects' coordinates. Moreover, most of the dataset's sign design is similar to the design used in a Russian Federation, which allows direct transfer of the learned behavior from our Avrora Unior robot's control system to the system of a real full-size autonomous car. We use OpenCV library that has built-in routines for cascade training. Training process consists of two stages – creating a training dataset and subsequent model training.

2.1. *Creating a Training Set*

A cascade can find desired objects faster and more accurately if these objects have same proportions and shapes. Therefore, all traffic signs were divided into two groups: signs of a triangular shape and signs of a round shape. We prepared two datasets with information about the signs for each group. Figures 1 and 2 show examples of each traffic signs group that were used in a training.

Fig. 1. Examples of round shape signs

2.2. *Cascade Training*

After a training set was constructed we are constructed a cascade of classifiers. Two training functions were considered: HAAR[12] (in honor of Alfred Haar) and LBP[13] (Local Binary Patterns). HAAR has shown to be in some cases more accurate (showing accuracy advantage of 10-20%), but in some cases it took several days to complete its training. LBP on the other hand requires significantly less training time – the training procedure took just several minutes. Yet, in a case of road signs, our tests did not detect any significant difference in accuracy between HAAR and LBP functions. It was decided to use LBP, since in case of the database changes, it would be possible to quickly re-train the cascade.

Kuranov[14] have shown that a Haar feature cascade with 20x20 sample size achieved the highest detection rate, while cascade of four 18x18 nodes was computationally more efficient with slightly inferior result. Figure 3 shows cascade detection results that was used with training images.

Fig. 2. Examples of triangle shape road signs

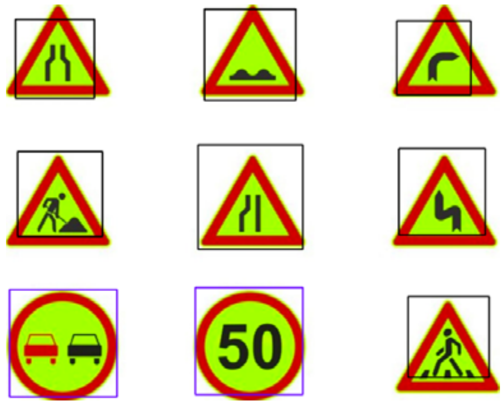

Fig. 3. Object detection by two cascades. The black contour is a cascade for triangular signs, the blue contour is a cascade for circular signs.

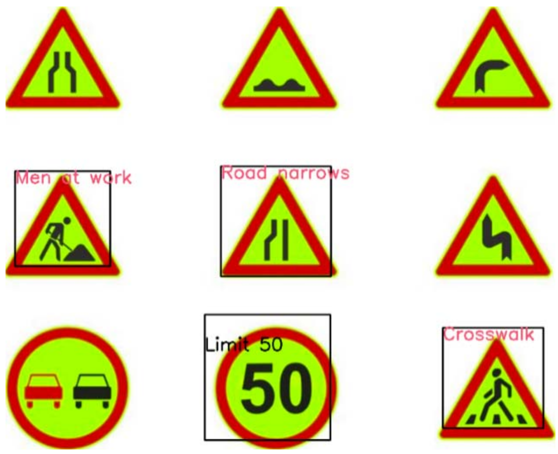

Fig. 4. Recognition of signs on a training image.

3. Algorithm Implementation

Avrora Unior is equipped with Microsoft Kinect, and since this camera's range is rather short, approximate dimensions of objects that the cascade could search for limits possible sign sizes. Our algorithm works in two stages. At the first stage ("LEARNING"), the algorithm uses one of available feature detector (ORB[15], SURF[16], SIFT[17]) to search for key features on training samples, and then stores results in a sign database. The second stage ("RECOGNITION") starts with a search of potential signs from a camera using our feature cascade, with results being passed to the classifier to determine an individual sign type. The comparison is performed using key feature comparators BFMatcher [18] and FLANN[19]. If a detected sign in a database has a relevant textual information it its displayed as an overlay on the frame (Fig. 4). Detected signs are saved into a separate database for further processing. Each of the cascades features an individual database, while each object from those databases is compared against samples for the training database.

Fig. 5. Sign models added into Gazebo simulator.

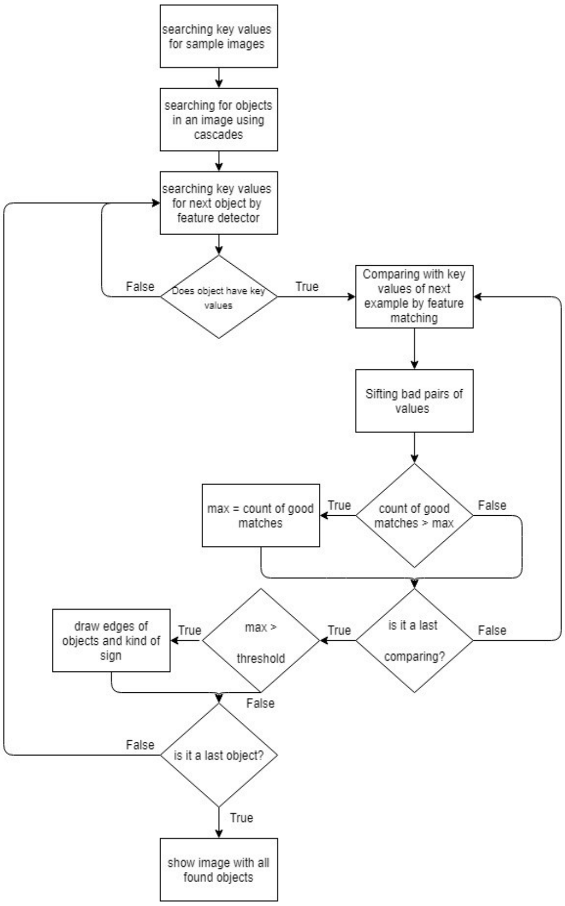

Fig. 6. Flowchart of a traffic sign recognition algorithm.

4. Algorithm comparison in Gazebo simulation

We performed validations in a simulated environment. For Gazebo simulator we created several models of signs of different type and stored them in SDF format. A set of traffic sign models was organized as a Gazebo model database[20] and a world was populated with the traffic signs (Fig. 5). that is used for Avrora Unior robot modeling to evaluate other parts of the autonomous control[12]. Our experimental design involved the following scenario:

- Avrora Unior robot drives in a straight line;
- Kinect camera captures frames to be used in traffic sign detection;
- ROS logging subsystem saves traffic sign recognition algorithm output of image processing, i.e., a detection status.

Table 1: Detection in different environments

Trial No.	Recognition time		
SIFT + BFMatcher			
	A	S	N
1	0.538	0.205	0.256
2	0.538	0.129	0.333
3	0.462	0.23	0.308
Average	**0.513**	**0.282**	**0.299**
ORB + BFMatcher			
	A	S	N
1	0.307	0.18	0.513
2	0.103	0.359	0.538
3	0.18	0.41	0.41
Average	**0.197**	**0.316**	**0.487**
SURF + FLANN			
	A	S	N
1	0.719	0.18	0.103
2	0.59	0.256	0.155
3	0.667	0.23	0.103
Average	**0.659**	**0.222**	**0.12**

The algorithm is presented in Fig. 6. A combinations of ORB with BFMatcher detectors and SIFT with BFMatcher, were tested in virtual experiments in Gazebo. Their accuracy in sign type recognition was lower than a combination of SURF with FLANN. The results are shown in Table 1: "A" means that it took less than 1 second to recognize a sign; "S" - it took more than 1 second to recognize a sign; "N" - a sign was not recognized. The first experiment took place in a regular environment illumination; the second experiment featured twilight-like environment with insufficient illumination; the third has featured excessive illumination. Road sign recognition algorithm failed to correctly recognize 4 out of 39 signs among the ones placed in the simulation world, i.e. had a 10.3% miss rate.

Algorithm	Detection rate
SIFT + BFMatcher	0.513
ORB + BFMatcher	0.197
SURF + FLANN	**0.659**

5. Conclusions and Future Work

In this paper we proposed a traffic sign recognition solution that uses several classifier cascades, one cascade for a triangular shape and one for a round shape. At the current stage the algorithm works poorly with narrow signs. Best accuracy in sign type recognition was demonstrated by a combination of SURF with FLANN.

In the future, we plan to add narrow sample images to training dataset. As a part of future work, we also plan to perform the same experiments in a laboratory environment with real car-like robot Avrora Unior.

Acknowledgements

This work was supported by the Russian Foundation for Basic Research (RFBR), project ID 18-48-160037.

References

1. Shabalina, K., Sagitov, A., Magid, E., Comparative Analysis of Mobile Robot Wheels Design. *International Conference on Developments in eSystems Engineering* (IEEE, 2018), pp. 175-179.
2. Greenhalgh, J., & Mirmehdi, M., Real-time detection and recognition of road traffic signs. *IEEE Trans. on Intelligent Transportation Systems* **13(4)** (2012), 1498-1506.
3. De Charette, R., & Nashashibi, F., Real time visual traffic lights recognition based on spot light detection and adaptive traffic lights templates. *Intelligent Vehicles Symposium* (IEEE, 2009), pp. 358-363.
4. Aufrere, R., Chapuis, R., & Chausse, F., A model-driven approach for real-time road recognition. *Machine Vision and Applications* **13(2)** (2001), 95-107.
5. Horn, B. K., & Schunck, B. G., Determining optical flow. *Artificial intelligence* **17(1-3)** (1981), pp. 185-203.
6. Shabalina K., Sagitov A.,Su K.L., Hsia K.H., Magid E., Avrora Unior Car-like Robot in Gazebo Environment. *International Conference on Artificial Life and Robotics* (2019), pp. 116-119
7. Magid, E., Lavrenov, R., Khasianov, A. Modified spline-based path planning for autonomous ground vehicle. *International Conference on Informatics in Control, Automation and Robotics* **2** (2017), pp. 132-141.
8. Klomp, M., et al., Trends in vehicle motion control for automated driving on public roads. *Vehicle System Dynamics* **57(7)** (2009), pp. 1028-1061.
9. Li, B., et al., Collaborative mapping and autonomous parking for multi-story parking garage. *IEEE Transactions on Intelligent Transportation Systems* **19(5)** (2018), pp. 1629-1639.
10. Dixit, S., et al., Trajectory planning and tracking for autonomous overtaking: State-of-the-art and future prospects. *Annual Reviews in Control* **45** (2018), pp.76-86.
11. Stallkamp, J., Schlipsing, M., Salmen, J., & Igel, C., The German Traffic Sign Recognition Benchmark: A multi-class classification competition. IJCNN **6** (2011), p. 7.
12. Soo, S., Object detection using Haar-cascade Classifier. Institute of Computer Science, University of Tartu, (2014) pp. 1-12.
13. Mu, Y., Yan, S., Liu, Y., Huang, T., & Zhou, B., Discriminative local binary patterns for human detection in personal album. *IEEE Conference on Computer Vision and Pattern Recognition* (2008), pp. 1-8.
14. Kuranov, A., An empirical analysis of boosting algorithms for rapid objects with an extended set of haar-like features. Intel Tech. Rep. (2002)
15. Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. R., ORB: An efficient alternative to SIFT or SURF. *ICCV* **11(1)** (2011), p. 2.
16. Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L., Speeded-up robust features (SURF). *Computer vision and image understanding*, **110(3)** (2008), 346-359.
17. Vedaldi, A. An open implementation of the SIFT detector and descriptor. UCLA CSD (2007).
18. Jakubovic, A. & Velagic, J., Image Feature Matching and Object Detection Using Brute-Force Matchers. *Int. Symposium ELMAR* (IEEE, 2018), pp. 83-86.
19. Goel, A., Saxena, S. & Bhanot, S., Modified functional link artificial neural network. *International Journal of Electrical and Computer Engineering* **1** (2006), pp. 22-30.
20. Lavrenov, R., Zakiev, A. Tool for 3D Gazebo map construction from arbitrary images and laser scans. In *10th International Conference on Developments in eSystems Engineering* (IEEE, 2017), pp. 256-261.