Avrupa Bilim ve Teknoloji Dergisi
Özel Sayı 22, S. 342-346, Ocak 2021

Araştırma Makalesi

www.ejosat.com ISSN:2148-2683

European Journal of Science and Technology
Special Issue 22, pp. 342-346, January 2021

Research Article

Convolutional Neural Networks Based Active SLAM and Exploration*

Akif Durdu[1], Nevzat Bol[1], Erol Öztürk[1], Mehmet Duramaz[1], Mehmet Korkmaz[2], Berat Yıldız[3†], Ahmet Kayabaşı[3]

[1] Konya Teknik Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Konya, Türkiye (ORCID: 0000-0002-5611-2322)
[2] Aksaray Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Aksaray, Türkiye (ORCID: 0000-0002-1462-8005)
[3] Karamanoğlu Mehmetbey Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Karaman, Türkiye (ORCID: 0000-0002-5675-6750)

(İlk Geliş Tarihi Aralık 2020 ve Kabul Tarihi Ocak 2021)

(**DOI:** 10.31590/ejosat.862953)

ATIF/REFERENCE: Durdu, A., Bol, N., Öztürk, E., Duramaz, M., Korkmaz, M., Yıldız, B. & Kayabaşı, A. (2020). Convolutional Neural Networks Based Active SLAM and Exploration. *European Journal of Science and Technology*, (22), 342-346.

Abstract

Mobile robots are high-performance robots that are used to perform a specific function in environments such as land, air and water, with free movement options and are equipped with many sensors for different processing capabilities. Today, it is used in many tasks such as object detection, tracking and mapping. Mobile robots used in mapping implementations are usually guided by user inputs. However, in some cases, this guidance is autonomously implemented through exploration algorithms that are examined under the active Simultaneous Localization and Mapping (SLAM) keyword. These algorithms are usually based on Laser Imaging Detection and Ranging (LIDAR) sensor. Since this sensor has a bulky structure and occupancy grid maps require heavy computing time, it is needed to develop new kinds of algorithms. In this study, we propose a novel Convolutional Neural Network (CNN) based algorithm that can create a map of an environment with a mobile robot that is independent of user inputs and move autonomously. For the first stage, the CNN structure is trained using the data set consisting of the environment image and the wheel angles related to these images so that the CNN model learns how to guide the robot. For the second stage, the robot is navigated autonomously through the trained network in an environment which is different from the first one, and the map of the environment is acquired simultaneously. Training and testing processes have been realized on a real-time implementation and the advantages of the developed method have been verified.

Keywords: Mobil Robot, Convolutional Neural Network (CNN), Simultaneous Localization and Mapping (SLAM), LIDAR, Active-SLAM.

Evrişimli Sinir Ağlarına Dayalı Aktif SLAM ve Keşif

Öz

Mobil robotlar, serbest hareket seçenekleri ile kara, hava ve su gibi ortamlarda belirli bir işlevi yerine getirmek için kullanılan ve farklı işleme kabiliyetleri için birçok sensörle donatılmış yüksek performanslı robotlardır. Günümüzde nesne algılama, izleme ve haritalama gibi birçok görevde kullanılmaktadır. Haritalama uygulamalarında kullanılan mobil robotlar genellikle kullanıcı girdileri tarafından yönlendirilir. Bununla birlikte, bazı durumlarda, bu yönlendirme, aktif Eşzamanlı Lokalizasyon ve Haritalama (SLAM) anahtar sözcüğü altında incelenen keşif algoritmaları aracılığıyla özerk olarak uygulanır. Bu algoritmalar genellikle Lazer Görüntüleme Algılama ve Değişim (LIDAR) sensörüne dayanır. Bu sensör hantal bir yapıya sahip olduğundan ve ızgara doluluk haritaları uzun bir hesaplama süresi gerektirdiğinden, yeni tür algoritmalar geliştirmek gerekir. Bu çalışmada, kullanıcı girdilerinden bağımsız olan ve otonom olarak hareket eden bir mobil robot ile bir ortamın haritasını oluşturabilen yeni bir Evrişimli Sinir Ağı (CNN) tabanlı algoritma öneriyoruz. İlk aşama için CNN modelinin robota nasıl rehberlik edeceğini öğrenmesi için çevre görüntüsü ve bu görüntülerle ilgili tekerlek açılarından oluşan veri seti kullanılarak CNN yapısı eğitilir. İkinci aşama için robot, birincisinden

* 1st International Conference on Computer, Electrical and Electronic Sciences ICCEES 2020 – 8-10 October 2020

† Corresponding Author: Karamanoğlu Mehmetbey Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Karaman, Türkiye, *beratyildiz@kmu.edu.tr

farklı bir ortamda daha önce eğitilmiş ağ üzerinden otonom olarak gezdirilir ve eş zamanlı olarak ortamın haritası alınır. Gerçek zamanlı uygulama üzerinden eğitim ve test süreçleri gerçekleştirilmiş ve geliştirilen yöntemin avantajları doğrulanmıştır.

Anahtar Kelimeler: Mobil Robot, Evrişimli Sinir Ağı (CNN), Eşzamanlı Lokalizasyon ve Haritalama (SLAM), LIDAR, Aktif-SLAM.

1. Introduction

There are many advantages to replacing human tasks with robots. In particular, robots capable of performing tasks in life-threatening areas have an important place in human life. Fukushima nuclear disaster is a striking instance of this situation where the need for replacing kamikazes with robots has been noticed clearly. Moreover, the use of robots instead of humans provides low-cost and effective labor.

One of the most fundamental characteristics of the task robots is of autonomous navigation capability. For a robot to be able to do autonomous navigation, information about where it is (localization), what the world around it looks like (mapping) and where it should be navigated (navigation) has to be given. For example, in an environment where the Global Positioning System (GPS) data are available, the robot can easily answer where it is. Similarly, the robot can recognize the world around it when the environment map is in existence. Besides, the robot knows where to go if task points are previously defined or the user guide is available. However, it may not be possible to access the GPS information in situations such as an indoor, tunnel. Hence, the localization problem has to be handled in another way. By the same token, in some situations, there is no environment map or a priorly obtained map may have changed for many reasons such as natural disasters. The environment map, therefore, should be regenerated or updated. As a result of the map deficiency, the robot task points may get changed. When all these problems are taken into account, the localization and mapping of the robot must be solved simultaneously and autonomous navigation have to be provided accordingly. In this context, there have been improved Simultaneous Localization and Mapping (SLAM) algorithms and significant engineering problems such as an autonomous driverless car can be cleared up through SLAM algorithms [1]. Although SLAM algorithms provide autonomy for mobile robots, full autonomy is possible if and only if the robot knows where to go. In other words, a robot that is aware of where it is, what the world around it looks like and where to navigate can have full autonomy. This definition is known as active-SLAM in the literature. Such an approach may require more effort, but it is crucial that a robot can decide where to go independent of human control [2].

Developing technological tools allow for heavy computation so that it is possible to go beyond classical machine learning techniques and benefit from deep learning algorithms that resemble human thought. Due to this power, many implementations which are originally done by machine learning techniques have been replacing nowadays with deep learning ones [3]. The study at hand is also related to the active-SLAM algorithm and it is based on Convolutional Neural Network (CNN) which is a special type of deep learning algorithm.

The novelty of the study is the combination of the SLAM scheme with the CNN based navigation instead of user guidance or existing active-SLAM algorithms. As to our knowledge, there is no comprehensive recent research of CNN for active-SLAM. The developed algorithm enables the idea of creating a map and autonomous navigation of many mobile robots which especially share the same environment.

2. Material and Method

2.1. Active SLAM

SLAM is an algorithm that a robot or robot team build a map of an unknown environment while simultaneously localize itself within this map. This algorithm generally has been introduced in two periods which are classical and modern. In the classical era, the problem has been solved with Bayesian-based filters. In that approach, a robot's odometry information and sensor data are applied to the filters such as Kalman or Extended Kalman Filters (EKF). In the early period of the SLAM researches, the EKF-based algorithms have matched excellently with the nonlinearity pattern of the SLAM. Therefore, EKF-SLAM has still regarded as an important cornerstone of the SLAM studies. Contrary to this, it is too slow when the number of landmarks in the environment is overabundant. This handicap is overcome with the development of the Rao-Blackwellised Particle Filter (RBPF-SLAM) which is based on particle filters. Generally, the inputs are the control (u) and sensor data (z) and the outputs are the locations of the robot (x) and a map of the environment (m), (1).

$$P(x_{1:k}, m \mid z_{1:k}, u_{1:k}) \qquad (1)$$

where k points out the steps of the whole algorithm. Whereas the classical period has consisted of Bayesian-based filters, the modern stage (also called Visual SLAM) is formed around computer vision algorithms. The environment map is created based on the images taken from cameras. There is a considerable amount of literature on V-SLAM using cameras such as monocular, stereo, RGBD [4].

To implement SLAM algorithms, there have been many open source software both for classical and modern approaches. Especially Robot Operating System (ROS) based ones such as gmapping, hectormapping, ORB-SLAM are the most preferred algorithms by researchers. This is because ROS provides a comprehensive outline for both real-time and simulation environments. It was also benefitted from the ROS hectormapping packages for the SLAM part of this study. This algorithm is based on a combination of the Laser Imaging Detection and Ranging (LIDAR) sensor data with the scan matching technique [5].

Regardless of the SLAM problem, the first autonomous exploration mentioned in the paper of Whaite in 1997 [6]. Later on, the techniques combined with SLAM has been improved. A general plan of these methods is the determination of the 2D occupancy grid map of an environment and afterward navigation of the robot to an unknown area. Furthermore, some studies measure the mapping uncertainty based on a pre-defined entropy and try to reduce it. Although these types of work have some upsides, the most drawback of them is that they are generally in need of an occupancy map of the environment to steer the robot.

A navigation problem can be solved by utilizing random movements or exploration algorithms. However, the random exploration process is not preferred due to both requiring high exploration time and having a risk of unexplored regions.

Because of this reason, the exploration algorithms which provide a systematic outline for the discovery of the whole region are preferred. A robot obtains a full autonomy and can navigate to the unknown regions of the environment without the need for user inputs by courtesy of these algorithms. In addition to that, active-SLAM is the integration of the exploration algorithms into the SLAM problem as mentioned in the introduction section. Thanks to the combined scheme, a robot can reach full autonomy.

2.2. Convolutional Neural Networks

So far, machine learning algorithms have been applied to various engineering problems. However, many papers reveal that such algorithms are still nowhere near human thinking. On the contrary, there are promising studies on deep learning that resembles human thought [7]. Because of that reason, a deep learning scheme is a good alternative to machine learning algorithms and have been applying to many fields of engineering problems. One of the secrets of the success of deep learning algorithms is big-size training data. Classical Central Processing Unit (CPUs) have trouble with the processing of big-size data due to time consumption. At this point, the Graphics Processing Unit (GPUs) is an excellent option. Enhanced GPUs and easy access to them have paved the way for the problem of processing big data and thus, researchers have begun to apply deep learning-based algorithms to the different applications. This paper also benefits from CNN which is a subset of deep learning algorithms. CNN is indeed an advanced ANN (Artificial Neural Network) and generally utilized in image classifications. The CNN structure used in the study illustrates in Fig.1.

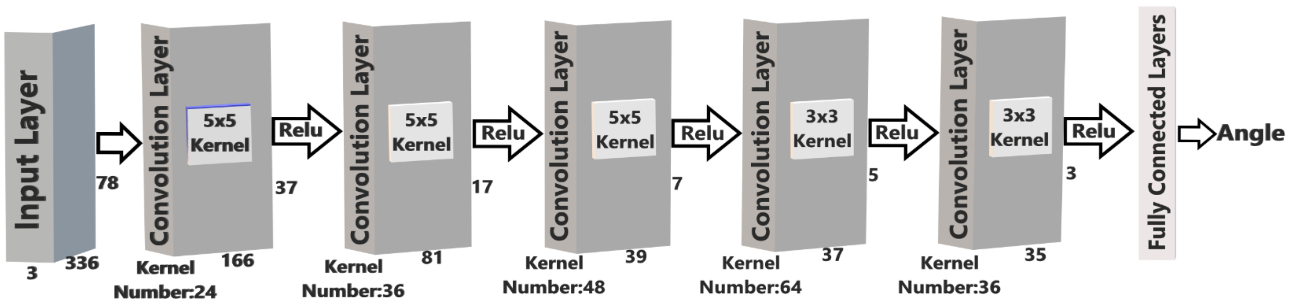


Fig. 1. Convolutional Neural Network architecture.

This structure consists of five convolution and four fully connected layers. A kernel with a size of 5x5 was used to feature extraction in the first 3 convolution layers and Kernel with a size of 3x3 were used in the last 2 layers. For these filtering operations 24, 36, 48, 64 and 64 Kernel were used respectively. After filtering, values were normalized using the Rectified Linear Unit (ReLu) activation function.

The dataset was generated with the images taken while driving and the wheel angles corresponding to these images. In other words, CNN input data is the vehicle and the output data is the wheel angles. The wheel angles in the collected data set in are in the range of [-0.35, 0.35] radians. Each image has a value which is corresponding to the steering angle at that time. The frequency of the steering angles for the whole process is illustrated in Fig. 2.

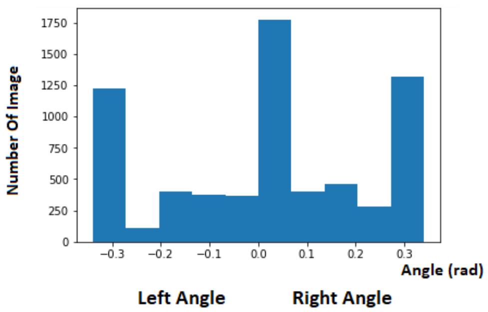


Fig. 2. Frequency of the steering angle.

The augmentation process is applied to diversify the data. The mixed data acquired from the images and angles is registered to the CNN structure. Frames of the images are multiplied with the randomized kernels and transformed into a single row. The low-sized filters obtained from the input images are applied to the three fully connected layers. The obtained output value evaluated through mean square error (MSE), (2) and backpropagation process is performed.

$$error = \frac{1}{n}\sum_{i=1}^{n}(\bar{X}_i - X_i)^2 \qquad (2)$$

3. Experimental Setup

A robot that has a differential steering model was used in the study. NVIDIA Jetson TX2 module, Stereolabs's passive zed camera and Scanse's sweep LIDAR were mounted on it. The Jetson module has a capacity of 8 GB RAM and 256 CUDA cores. LIDAR has 360 degrees field of view and its sample frequency is 2 to 8 kHz in Fig. 3.

Fig. 3. The robot used in the experiment.

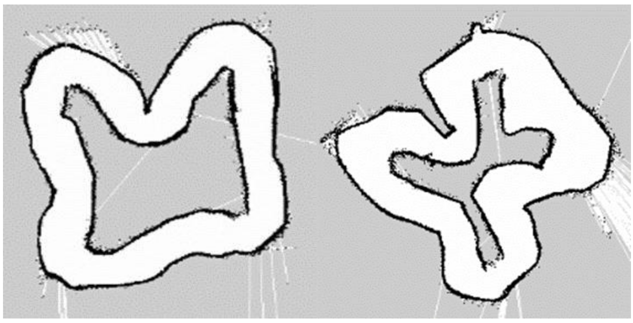
Within the scope of the application, the environment where the robot carries out the training and testing stages is created as shown in Fig. 4 in a way similar to the environments in hospitals, schools, etc. The training data which consists of environment images and joystick commands (also meaning to steer angle) was collected as the robot was steering in this environment in Fig. 4a. The created CNN architecture was trained with these inputs. And later, this network was tested in a different environment in Fig. 4b.

(a)
(b)

Fig. 4. The environment of the experiment. (a) The training environment. (b) The test environment.

4. Results and Discussion

According to the test results, it was witnessed that the robot was able to move autonomously in this unknown test environment. During the test process, neither user data nor information about how to navigate was provided to the robot and all the movements were realized by the trained network. The 2-D occupancy grid map of the training environment is seen in Fig. 6a. In the process of autonomous movements, the test environment map was obtained using the ROS hectormapping package along with the laser sensor mounted on the robot in Fig. 6b.



(a) (b)

Fig. 6. Maps of the environments acquired by robots. (a) The training environment 2-D map. (b) The test environment 2-D map.

Furthermore, it also uses the signs Quick Response (QR) codes to determine which path should be selected at the points where alternative routes are available. To realize this, the ID and priority path information were assigned to each turning point by the signs (QR codes) in Fig. 7. This information was logged during the experiments. Thanks to the logged information, when the robot revisits any turning point, the control algorithm was guided the robot in the other direction which was not visited yet.

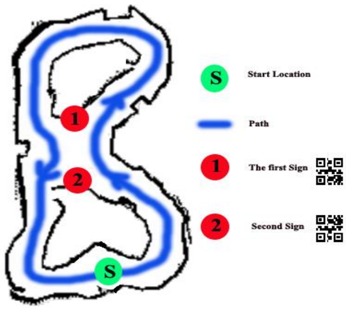


Fig. 7. Quick response codes.

Consequently, within the framework of active-SLAM, a robot was built the environment map and at the same time was able to find an answer to the question of where to steer using a trained CNN scheme. The best result parameters for system performance are given in Table I in the training and test procedures where the CNN structure is used.

Table 1. Deep Learning Parameters

Parameters	**Numbers**
Number of Training Image	*5376*
Number of Testing Image	*1344*
Batch Size	*8*
CNN Activation Function	*ReLu*
CNN Loss Function	*Mean Squared Error*
CNN Optimizer	*Adam*
Epochs	*40*

5. Conclusions and Recommendations

In this study, SLAM algorithms were combined with the autonomous exploration methods and an active-SLAM framework for this scheme was constituted. Although there are several methods for autonomous navigation of a mobile robot, it is seen that most of them are generally related to the determination of the unknown regions and steering of the robot into these areas. In our work, the CNN scheme that recently received much attention by many researchers was unified with the SLAM algorithm. The CNN framework was trained with real-time environment images and the steering angle of the robot. To verify the trained network, a new test environment was made up and real-time experiments were carried out on it. Although the test and training environments were not the same, it was observed that the robot made rotation and forward motion decisions by itself and it successfully fulfilled the autonomous navigation. As well as autonomous navigation, the mapping of the environment was performed simultaneously, thereby the robot gained full autonomy. The enhanced method was tested and verified with real-time implementations. Utilizing the artificially created QR codes, the robot can explore the whole environment and does not have to visit the previously observed regions. In future studies, it is planned to expand the algorithm by adding feature detection modules such as SIFT, SURF, ORB so that it will not need artificial landmarks. On the other hand, it is also planned to investigate situations where the environment has open/closed sections or certain indicators such as direction signs (stop sign, crosswalk, etc.). To further our research, we intend to develop the study by adding the action recognition part. Thanks to action recognition, it is likely to present an active-SLAM scheme that socially incorporates with humans.

6. Acknowledge

Authors are thankful to Rac-Lab (www.rac-lab.com) for providing the trial version of their commercial software for this study.

References

[1] Durrant-Whyte, H., & Bailey, T. (2006). Simultaneous localization and mapping: part I. *IEEE robotics & automation magazine*, 13(2), 99-110. Available: http://dx.doi.org/10.1109/MRA.2006.1638022

[2] Maurović, I., Seder, M., Lenac, K., & Petrović, I. (2017). Path planning for active SLAM based on the D* algorithm with negative edge weights. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(8), 1321-1331. Available: http://dx.doi.org/10.1109/TSMC.2017.2668603

[3] Zeng, Z., Xiao, H., & Zhang, X. (2016). Self CNN-based time series stream forecasting. *Electronics Letters*, 52(22), 1857-1858. Available: http://dx.doi.org/10.1049/el.2016.2626

[4] Fuentes-Pacheco, J., Ruiz-Ascencio, J., & Rendón-Mancha, J. M. (2015). Visual simultaneous localization and mapping: a survey. *Artificial intelligence review*, *43*(1), 55-81.

[5] Kohlbrecher, S., Von Stryk, O., Meyer, J., & Klingauf, U. (2011, November). A flexible and scalable slam system with full 3d motion estimation. In *2011 IEEE international symposium on safety, security, and rescue robotics* (pp. 155-160). IEEE. Available: http://dx.doi.org/10.1109/SSRR.2011.6106777

[6] Whaite, P., & Ferrie, F. P. (1997). Autonomous exploration: Driven by uncertainty. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*(3), 193-205.

[7] Fatih, Ö. (2019). Efficient deep feature selection for remote sensing image recognition with fused deep learning architectures. *The Journal of Supercomputing*, vol. 76, no. 11, pp. 8413–8431, 2020. DOI: 10.1007/s11227-019-03106-y.