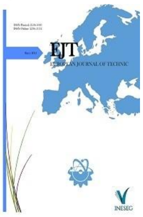
European Journal of Technique

journal homepage: https://dergipark.org.tr/en/pub/ejt

Vol.14, No.1, 2024



Research Article

Text Detection and Recognition in Natural Scenes by Mobile Robot

Erdal Alimovski[1*], Gökhan Erdemir[2] and Ahmet Emin Kuzucuoğlu[3]

[1*] Istanbul Sabahattin Zaim University, Computer Engineering Department, 34303, Istanbul, Turkiye. (e-mail: erdal.alimovski@izu.edu.tr).
[2] University of Tennessee at Chattanooga, Dep. of Engineering Management and Tech., Chattanooga, TN 37405, USA. (e-mail: gokhan-erdemir@utc.edu).
[3] Marmara University, Department of Electrical and Computer Engineering, 34722, Istanbul, Turkiye. (e-mail: kuzucuoglu@marmara.edu.tr).

ARTICLE INFO

Received: Dec., 19. 2023
Revised: Apr., 17. 2024
Accepted: Apr, 20. 2024

Keywords:
Natural scene
Mobile robot
Optical character recognition
Text detection
Text recognition

Corresponding author: *Gökhan Erdemir*

ISSN: *2536-5010* | e-ISSN: *2536-5134*

DOI: https://doi.org/10.36222/ejt.1407231

ABSTRACT

Detecting and identifying signboards on their route is crucial for all autonomous and semi-autonomous vehicles, such as delivery robots, UAVs, UGVs, etc. If autonomous systems interact more with their environments, they have the ability to improve their operational aspects. Extracting and comprehending textual information embedded in urban areas has recently grown in importance and popularity, especially for autonomous vehicles. Text detection and recognition in urban areas (e.g., store names and street nameplates, signs) is challenging due to the natural environment factors such as lighting, obstructions, weather conditions, and shooting angles, as well as large variability in scene characteristics in terms of text size, color, and background type. In this study, we proposed a three-stage text detection and recognition approach for outdoor applications of autonomous and semi-autonomous mobile robots. The first step of the proposed approach is to detect the text in urban areas using the "Efficient And Accurate Scene Text Detector (EAST)" algorithm. Easy, Tesseract, and Keras Optical Character Recognition (OCR) algorithms were applied to the detected text to perform a comparative analysis of character recognition methods. As the last step, we used the Sequence Matcher to the recognized text values to improve the method's impact on OCR algorithms in urban areas. Experiments were conducted on the university campus by an 8-wheeled mobile robot, and a video stream process was carried out through the camera mounted on the top of the mobile robot. The results demonstrate that the Efficient And Accurate Scene Text Detector (EAST) text detection algorithm combined with Keras OCR outperforms other algorithms and reaches an accuracy of 91.6%

1. INTRODUCTION

In recent years, Computer Vision (CV) technology has made remarkable progress in practice and has a wide range of applications across many different fields, such as robotics [1], automotive[2], healthcare[3], agriculture [4], etc. Mobile robots, which are increasingly used daily, have started to be used frequently, especially in work such as delivery in urban areas. Therefore, they can be operated in different places, like streets, homes, hospitals, and factories. As a result of the interaction of mobile robots with their working environments and the need to sense some environmental features or landmarks, CV-based algorithms have become common for indoor and outdoor applications. Environmental knowledge can help mobile robots position themselves, alter their movement, and make decisions without human intervention [5]. Therefore, extracting rich semantic information such as textual information, objects and structures, terrain features, and landmarks from the environment is necessary. For example, if the mobile robot operates in urban areas, it can benefit from textual information about street names, shop names, and signboards. In this way, it may navigate itself according to the obtained data.

Text detection and recognition in urban areas (e.g., streets, squares, university campuses) has drawn much interest and is essential in various computer vision-based applications. Performing detection and recognition of text in urban areas is difficult because of the variety of backgrounds, low resolution, font types, distortion, occlusions, etc. [6]. Text detection, a prerequisite of text recognition, is a critical phase in textual information extraction and understanding. The fundamental text detection component is designing features that differentiate text from the backgrounds. In traditional methods, features are manually intended to acquire the characteristics of scene text [7], [8], whereas features are obtained entirely from training data in deep learning-based techniques [9], [10]. Generally, traditional methods comprise sliding window-based and connected components-based methods. To detect text, sliding window-based techniques change a window at each position in an image [11]. Character candidates are initially extracted as a first step in connected components-based approaches, and then post-processing is done to remove non-text noise and connect the candidate [12]. In [13], [14], authors performed (HOG) and Random [14] Ferns to identify the characters. Following through pictorial structure, a particular word's optimal configuration was found. Mishra et al. utilized sliding windows and a Conditional Random Field model with a combination of bottom-up and top-down cues to identify character candidates [15]. A part-based tree-structured model was created in the study [14] to identify the characters in cropped photos. Authors in [16] proposed a combination of multi-scale mid-level features called Strokelets as an alternative for character representation. However, the efficiency of traditional methods is constrained in problematic conditions where low resolution, multi-orientation, and perspective distortion exist.

Recently, convolutional neural networks (CNNs) have been widely applied to text detection tasks to overcome the limitations of traditional methods. The Rotation Region Proposal Network (RRPN) was introduced, offering new Faster-RCNN components [17]. The proposed framework aims to detect arbitrary-oriented text in various environments.

The framework is designed to produce inclined proposals with text orientation angle data, which later is utilized for bounding box regression. Therefore, to set the arbitrary orientation proposals, the RoI pooling layer is introduced [18]. In [19], authors proposed a model that employs a Fully Convolutional Network (FCN) and one-step Non-Maximum Suppression (NMS) structure for scene text detection. FCN model consists of three main parts: feature extraction, feature fusion, and multi-task learning. Localizing the scene text's quadrilateral boundaries is adequate due to direct regression. Results from experiments have demonstrated the efficiency of the proposed model. In another study [19], authors proposed a novel model to detect arbitrary-oriented texts called Rotation Region CNN, based on Faster-RCNN. First, the Region Proposal Network (RPN) generates axis-aligned bounding boxes. Axis-aligned bounding boxes are refined and inclined minimum area boxes are predicted using pooled features. Finally, to obtain the detection results, NMS is performed.

OCRs have demonstrated impressive performance in various commercial applications over the past few decades, focusing on natural scene texts [20]. The ability of OCR technology to recognize documents with a constant background color, the most basic fonts, and nicely aligned text is impressive. However, the performance in scene text recognition, such as bills, traffic signs, and shop names, is limited due to the complex backgrounds, distinct and distorted fonts, uneven illumination, and color variations [21]. Consequently, scene text recognition has grown in popularity as a field for CV applications in robots. Character-based and full word-based recognition are the two main categories of conventional text recognition methods used in urban areas. For scene text recognition, many research efforts have been carried out. For example, authors in [22] proposed detecting and recognizing scene text frameworks for multi-oriented texts. Forest classifiers demonstrated tasks of text detection and recognition. Comprehensive experiments show that the suggested algorithm performs better than the existing approaches. In [23], a model was presented to perform recognition of scene texts under different orientations, such as vertical texts, top-to-bottom and vice versa, horizontally stacked vertical texts. The proposed model comprises three main processes: text localization, segmentation, and recognition. For localization and segmentation, excluding the pre-processing and post-processing steps, they used the maximally stable extremal regions detector while recognizing the Tesseract algorithm was performed. The results show the proposed model's effectiveness for the vertical text recognition task. In natural scenes, Ebin Zacharias et al. mounted a camera in a vehicle to detect and recognize [24]. Gamma correction, skew correction, and canny edge detector were performed to get the region of the text in an image. Further, Tesseract 5 with Long-Short-Term Memory (LSTM) was utilized during the recognition phase. This applied approach achieved around 83% correct character recognition rate.

In this study, we propose an approach that automatically recognizes the text information in signboards on the robot's working areas during mobile robot movement. The EAST algorithm was performed to detect the text values. Once the textual pattern was detected, the detected patterns were transmitted to Easy, Keras, and Tesseract OCR algorithms to compare character recognition methods in urban areas. In addition, we implemented a word similarity calculator method called the Sequence Matcher (SM) to repair wrongly recognized words and to examine its effect on OCR algorithms in urban areas.

2. MATERIALS AND METHODS

The proposed approach to detect and recognize the textual patterns in the urban areas during the mobile robot's movement consists of three phases: data collection, mobile robot control, and applied algorithms. Data collection was performed on campus with an 8-wheeled mobile robot controlled remotely. After investigating the existing literature, the EAST algorithm was used for text detection. As a first step, EAST detected the textual patterns in the live video. After that, it was transmitted to Easy OCR, Keras OCR, and Tesseract OCR for text recognition. Thus, the performance of the open-source OCR algorithms in urban areas on a live video stream was compared and evaluated by comprehensive experiments in this study. Figure 1 shows the study flowchart.

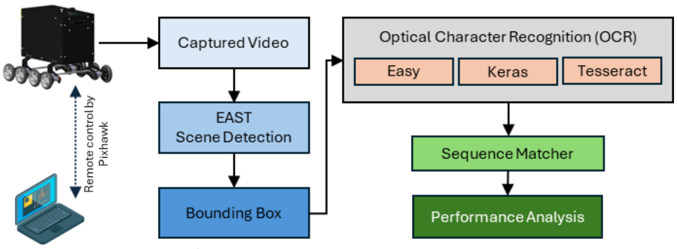


Figure 1. Flowchart of the proposed approach.

2.1. Data Collection

As mentioned in the introduction, more research is needed to recognize text in urban areas, such as streets, squares, etc., using open-source OCR algorithms. Therefore, this research captured live videos on campus utilizing a Zed2i camera [25] mounted at the top of an 8-wheeled mobile robot, where details will be clarified in the next section. In the conducted scenario, start and finish points were defined for the mobile robot. Thus, data were gathered in a specific area on campus where signboards, shops, and stationery are often used.

In addition, the fact that the signboard texts in that region were easily recognizable by the human eye played an important role in choosing that region. Thus, in the testing phase, we could compare the recognized text by the algorithm with the actual text. Figure 2 presents the satellite image of the region, which contains the locations of each signboard on the test location. Also, the following path of the mobile robot is presented in the same figure. Pixhawk controller[26] was used for trajectory tracking and control of the robot. The existing text on the test area signboards is indicated in Table I.

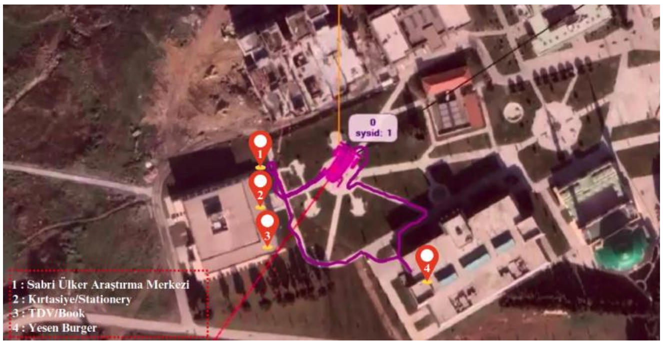


Figure 2. Area of data collection phase.

TABLE I
LABELS IN SIGNBOARDS

Signboard Nr	Signboard Name	Existing Label
1	Sabri Ülker Araştırma Merkezi	"Sabri", "Ülker", "Araştırma ", "Merkezi"
2	KIRTASİYE STATIONARY	“KIRTASİYE”, “STATIONARY”
3	TDV Book Store Kitabevi	"TDV","Book", "Store","Kitabevi"
4	Yesen Burger	"Yesen", "Burger"

As seen in Figure 3, each signboard contains different textual patterns with fonts, backgrounds, colors, and board types. For example, some shop names' characters have various colors and font types. All tests were performed under limited lighting conditions during cloudy weather. During experiments, the camera's resolution was 3840x1080 pixels, with 30 FPS, while the total video record duration was around 35 minutes.

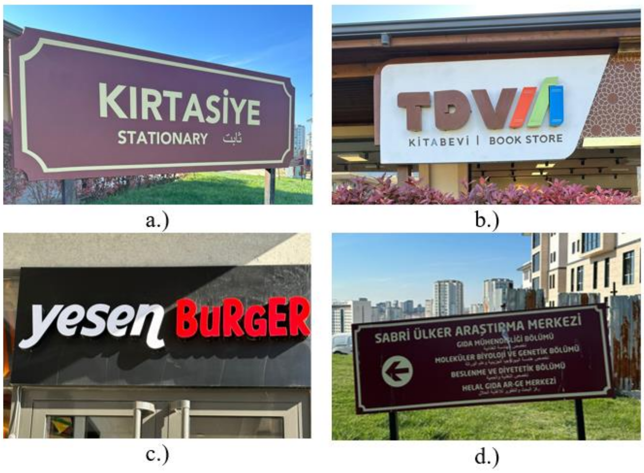


Figure 3. Signboards samples in the test area.

2.2. Mobile Robot Platform

The 8-wheel drive mobile robot designed in [27] was used during this study's data collection and experiments. It has eight wheels that can climb the pavements. The robot's dimensions are listed as total weight: 20 kg, width: 648 mm, length: 640 mm, height: 571 mm, and tire diameter: 140 mm [27]. In the standard configuration of the robot, the camera was mounted on the top of the robot. For the default setup of the robot, the camera's position was out of angle concerning target views for detection. This situation could hurt the applied text detection and recognition algorithms. The platform was designed and mounted on the top of the mobile robot to bring the camera to the same angle as the target views. Thus, the camera was set 1.13 m above the ground after the modifications. The mobile robot, after the modification, is presented in Figure 4.

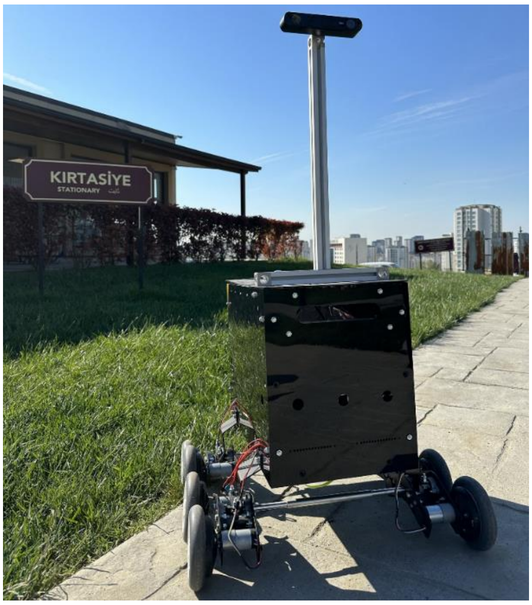


Figure 4. Mobile robot with vision system.

2.3. Camera and Controller

The Zed2i camera was used in this study to gather visual data from the working environment. Pixhawk controller was used to control and track the robot's trajectory. The Zed2i camera and Pixhawk controller have often been used in various robotic applications. While the Zed2i camera can obtain 3D images and depth information due to its stereo imaging technology, the Pixhawk controller provides significant convenience in autonomous driving in robotic systems such as unmanned aerial vehicles (UAV) and unmanned ground vehicles (UGV). Stereolabs developed a ZED2i camera for image acquisition in various applications such as virtual reality, augmented reality, robotics, industrial automation, and many others. It is suitable not only for indoor but also for outdoor applications. The camera has a robust and reliable IR sensor that can provide high-quality images even in low-lighting conditions. The ZED2i captures both RGB and depth images in high resolution (2K) with passive stereoscopic 3D technology based on a composite stereo image of the camera. The ZED2i camera is compatible with NVIDIA Jetson platforms and other ARM-based systems. This makes it appropriate for various applications, including mobile robots, UAVs, self-driving vehicles, and other intelligent devices.

Pixhawk is an open-source autonomous vehicle control system. It can provide a wide range of applications for robotic systems when integrated with ArduPilot software. Pixhawk comprises various modules, such as GPS, high-speed processors, and configurable interfaces. It can accomplish the navigation and trajectory control tasks crucial for mobile robots. To get the actual position of the mobile robot and perform trajectory tracking reliably, the u-blox NEO M8N GPS Module was mounted to the Pixhawk board. The GPS module starts searching for satellites and establishes a connection after a few minutes of being connected to the Pixhawk. Then, it sends the received location data packages to the board. Thus, the remote-control station can wirelessly receive the location data thanks to 433 MHz Telemetry Radio modules. Mission Planner (MP) software was installed on the remote-control station. Telemetry modules established the connection between the Pixhawk board and the remote PC. Hence, the robot was controlled and tracked by MP.

2.4. Scene Text Detection

Scene detection algorithms use image processing and artificial intelligence techniques to detect text accurately and effectively in complex scenes. These algorithms are used in many applications to detect and recognize, such as signage, logos, traffic signs, and security systems. Towards the literature review of scene detection algorithms, the EAST algorithm used in this study surpasses other algorithms in obtaining fast and accurate results. Therefore, it was decided to be used.

Scene text detection techniques have been accomplished on various benchmarks. These techniques, especially those that employ deep neural network models, have limitations when dealing with complex scenes. Interactions between the modules in the algorithmic model impact the text detection models' performance. Thus, text detection performance can be improved by utilizing a basic model that optimizes the loss function. Therefore, using a simple and effective EAST algorithm, text regions can be detected more rapidly and accurately [28]. The EAST algorithm was proposed by Zhou in 2017 to surpass the limitations in scene detection. The EAST algorithm employs a single neural network to predict the text's words or lines by avoiding candidate aggregation and word segmentation.

The model can eventually predict the 1-channel score map and the 4-channel box map if box data are tagged as RBOX. The algorithm can ultimately estimate the 1-channel score map and the 8-channel BOX map if the box data are annotated as QUAD.

2.5. Character Recognition

As a first step, the EAST algorithm detected the text regions and taken into a bounding box. Then, the detected pattern was transmitted to three open-source OCR algorithms. These are Easy OCR, Keras OCR, and Tesseract OCR. In this study, the performances of these character recognition algorithms were compared.

2.5.1. Easy OCR

Easy OCR is a cutting-edge OCR library that performs efficiently and supports over 80 languages. It was developed using Python and PyTorch frameworks. It executes detection using the CRAFT [29] method, a scene text detection model based on neural networks.

As mentioned previously, the EAST algorithm was performed for scene detection in this study. Therefore, both EAST and CRAFT algorithms were used for text detection before feeding the detected texts to the recognition phase. For recognition, Easy OCR uses a Convolutional Recurrent Neural Network (CRNN) [30], which is comprised of three parts: ResNet for feature extraction [31], LSTM sequence labeling [32], and Connectionist Temporal Classification (CTC) decoding [33]. The Easy OCR engine is among the best because of the pre-processing procedures included in this pipeline.

This OCR engine can identify and detect more than 80 languages. We employed the algorithm in Turkish configuration since our dataset contains Turkish texts. When the text is detected and taken into the bounding box, we feed it to the OCR engine to recognize the texts in the environment.

2.5.2. Keras OCR

Keras OCR is a deep learning-based OCR method in the Keras library [34]. Keras OCR offers a convenient interface for developing OCR models, which can identify text in various formats, including handwritten, printed, and even noisy or damaged text. Keras OCR has many typical applications, such as digitizing article texts, scanning, processing financial documents, and detecting texts in traffic signs, streets, and shops. Keras OCR is composed of two neural network architectures: CRAFT and CRNN. By investigating each character region and the affinities between characters, Keras OCR employs CRAFT to detect the text areas. Whereas for the text recognition phase, the original CRNN model was used. Due to the high accuracy of EAST text detection and some noisy detections of CRAFT during the implementation, we decided to disable the detection part of Keras OCR and utilize it just for character recognition.

2.5.3. Tesseract OCR

Tesseract is a well-known open-source OCR engine that Hewlett-Packard first developed and later supported by Google. Tesseract has been adapted for more than 140 different languages [35]. Since the 4.0 version, a new engine built on Long Short-Term Memory (LSTM) was developed. Compared to prior versions of Tesseract, LSTM, a particular type of RNN, offers significantly improved accuracy. Additionally, the "pytesseract" Python library exists and offers quick access to this engine in Python. The image was initially transformed into a binary image using an adaptive threshold. Following, character outlines were extracted by applying connected component analysis. Next, to organize the outlines into words, methods for character split and character association are performed. The two-password recognition process is ultimately carried out using clustering and classification approaches. To make its final determination regarding the recognized word, Tesseract consults both the language and user-defined dictionaries. Thus, the word with the smallest distance is given as an output.

2.6. Sequence Matcher

Sequence Matcher (SM)[36] is a class of ‘difflib’ modules used to compare the similarity of two given strings. The Ratcliff/Obershelp algorithm [37] is run in the background. After comparing the two strings, the algorithm returns a score between 0 and 1. After comparing two strings, if the obtained score is greater than 0.7, it will be regarded as a keyword and stored as an actual word or label. The equation of the algorithm is as follows:

$$D_{ro} = \frac{2 * K_m}{|S1||S2|} \tag{1}$$

where K_m represents the number of the same characters in sequence, whereas $|S1|$ and $|S2|$ give the corresponding length for each of these two strings.

3. EXPERIMENTAL STUDY

In this study, once the video data was gathered with the mobile robot, the experiments were carried out on the computer with an Intel i5 Central Processing Unit (CPU), 16 GB random access memory (RAM), and a single Graphics Processing Unit (GPU) NVIDIA GeForce GTX 1680.

The EAST algorithm was first performed during the experiments to detect the target scene texts. Thus, textual patterns in signboards, shop boards, and others have been taken into a bounding box. Secondly, the detected textual pattern was inputted into Easy OCR, Keras OCR, and Tesseract OCR algorithms for the character recognition step. To evaluate the performance of the OCR algorithms, we define the accuracy metric, the number of correctly recognized words divided by the total number of existing words multiplied by 100\%. In addition, an analysis was conducted regarding how many times the existing words were recognized correctly by each algorithm and recognition time. Besides, the effect of the SM algorithm over OCR algorithms was investigated.

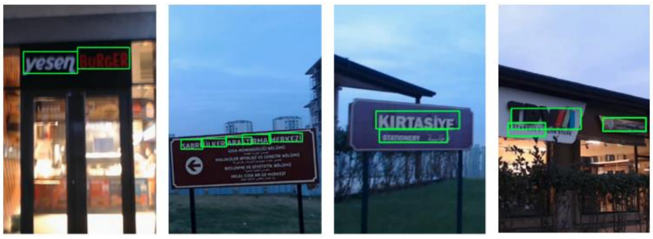


Figure 5. A sample of textual pattern detection by the EAST algorithm in cloudy weather.

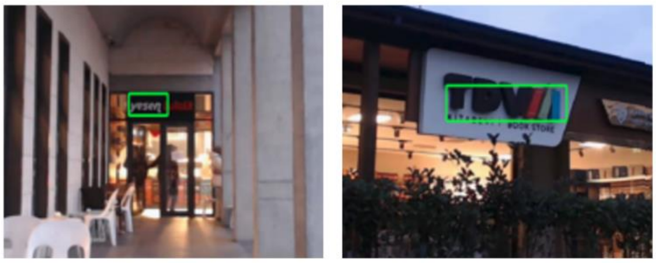


Figure 6. A sample of textual pattern detection by the EAST algorithm under limited lighting conditions.

In most cases, the EAST algorithm accurately detected the textual patterns in signboards, except in cases where the camera's distance was far away from the possible target texts. Its performance in cloudy weather and under limited lighting conditions is highly accurate, improving OCR algorithms' performance and correctness. In Figure 5, several boards with text that are accurately detected are presented. During the experiments, it was observed that when the angle of the mobile robot's camera was on the side position, the EAST algorithm was limited in detection. Therefore, as shown in Figure 6, the alignments of the bounding box vary.

When Table II is examined, it is clear that the Keras OCR algorithm outperformed other algorithms in accuracy. In addition, the average recognition time of the Keras OCR algorithm was the shortest compared to the other OCR algorithms. The algorithm recognized all the labels in the scene except the label "araştırma" even in bad conditions, such as low illumination and blurriness on the video stream caused by the vibrations of the robot. In contrast, the ability of the Easy OCR and Tesseract OCR to recognize the labels in urban areas was limited. Thus, both algorithms could not recognize several labels such as "TDV" and "Burger".

TABLE II

PERFORMANCE ANALYSIS OF OCR ALGORITHMS.

Method name	Average Recognition Time (second)	Accuracy (%)
Easy OCR	0.06	83.3
Keras OCR	0.04	91.6
Tesseract OCR	0.08	75

On the other hand, we measured the performance of the algorithms in terms of the number of repeats of each label and the recognition time of each label, as listed in Table III. Notice that recognition time means the time passed between text detection and recognition.

Table III demonstrates that the Keras OCR algorithm recognized each label at least two or three times more than Easy OCR and Tesseract OCR. Besides this, recognition time is less than other algorithms for each label. Some labels, such as *Sabri*, *Kırtasiye*, and *Yesen*, were recognized many times, while labels such as *Stationary*, *TDV*, *Store*, and *Kitabevi* were less frequently identified. In some cases where the mobile robot was moved on a smooth ground surface, the vibration was lesser. Thus, the video quality was good, so the algorithms performed better due to the camera position alignment and type of surfaces. On the other hand, as the mobile robot was moved across the grass, the vibrations increased, and the video quality became blurry, making it harder for the algorithms to recognize the labels.

During the experiments, it was observed that character recognition algorithms misidentified some labels by missing one or two characters of the actual labels. For example, algorithms recognized the actual labels that are *sabri*, *ülker*, *merkezi*, *Book*, and *Store* as ***abri***, **ülke**, ***merkez***, ***boo***, and ***tore***, respectively. It is caused by the blurriness of the video stream and the camera's position. It was worth noting that, during the movement of the mobile robot, some vibrations occurred due to the ground type, affecting the quality of the video stream.

We applied the SM method to address the shortcomings in the previous step, which calculates the similarity between two words. The SM supported the output of each OCR algorithm, and it figured out the word similarity between the recognized label and actual labels in a text file, which was previously stated. Thus, it suggested the most appropriate words. In our case, we set a threshold of 70%, so it only suggested labels higher than that percentage. Table IV demonstrates the performance results of each algorithm after applying the word similarity calculation method. Compared with Table III, it was observed that each algorithm's recognition performance increased for all the labels.

TABLE III
PERFORMANCE COMPARISON OF OCR ALGORITHMS

	Easy OCR		**Keras OCR**		**Tesseract OCR**	
Recognized Labels	Rec. Number	Average Time (s)	Rec. Number	Average Time (s)	Rec. Number	Average Time (s)
Sabri	10	0.05	16	0.04	3	0.08
Ülker	7	0.08	4	0.04	2	0.07
Araştırma	-	-	-	-	-	-
Merkezi	7	0.06	9	0.03	1	0.08
Kırtasiye	63	0.08	69	0.04	19	0.08
Stationery	1	0.07	4	0.03	1	0.07
TDV	-	-	4	0.03	-	-
Book	1	0.09	2	0.04	1	0.08
Store	1	0.06	2	0.05	1	0.09
Kitabevi	1	0.07	3	0.04	1	0.09
Yesen	16	0.10	160	0.05	75	0.08
Burger	2	0.08	71	0.05	-	-

TABLE IV
PERFORMANCE COMPARISON OF OCR ALGORITHMS WITH THE SEQUENCE MATCHER (SM) METHOD

	Easy OCR & SM	**Keras OCR & SM**	**Tesseract OCR & SM**
Recognized Labels	Recognition Nr.	Recognition Nr.	Recognition Nr.
Sabri	28	39	11
Ülker	19	16	7
Araştırma	6	7	1
Merkezi	20	27	4
Kırtasiye	89	99	35
Stationery	8	14	7
TDV	5	13	3
Book	7	9	2
Store	4	8	5
Kitabevi	8	10	4
Yesen	31	246	115
Burger	6	127	4

Table IV demonstrates the performance results of each algorithm after applying the word similarity calculation method. The experiments show that applying word similarity methods positively affects character recognition algorithms, especially in limited labels.

After examining the experiments, it can be concluded that Keras OCR combined with the EAST algorithm performs well in urban areas scenarios. However, Easy OCR and Tesseract OCR have some recognition and time limitations.

4. CONCLUSIONS

Extracting textual information in urban areas is vital because it includes crucial environmental information for robot navigation. The dataset was collected on the university campus with a camera mounted at the top of the mobile robot. As a first step, the EAST algorithm detected textual patterns in natural scenes. Secondly, the detected textual patterns were transmitted to three OCR algorithms, Easy OCR, Keras OCR, and Tesseract OCR, for the character recognition step. As a result, the performance of these OCR algorithms was tested, analyzed, and compared. Performance analysis of each OCR algorithm was conducted based on correctly recognized labels and recognition time. Additionally, we added the Sequence Matcher method to OCR algorithms to analyze its effect on OCR algorithms in urban areas. While capturing video from the camera during the robot's movement, the viewing angle was not always sufficient to read the entire text on the signboards; thus, some labels were recognized as missing one or two characters. However, by applying a fusion of OCR algorithms and the Sequence Matcher method, this problem has been eliminated, and the number of label recognition has increased significantly. When all experiments are investigated, the obtained results show that the Keras OCR algorithm achieved the highest results compared to the Easy OCR and Tesseract OCR algorithms in terms of accuracy, recognition number of labels with and without the Sequence Matcher method, and recognition time.

REFERENCES

[1] D. Sankowski and J. Nowakowski, *Computer Vision in Robotics and Industrial Applications*, vol. 3. WORLD SCIENTIFIC, 2014. doi: 10.1142/9090.

[2] M. Nagy and G. Lăzăroiu, “Computer Vision Algorithms, Remote Sensing Data Fusion Techniques, and Mapping and Navigation Tools in the Industry 4.0-Based Slovak Automotive Sector,” *Mathematics*, vol. 10, no. 19, 2022, doi: 10.3390/math10193543.

[3] M. Bicakci, O. Ayyildiz, Z. Aydin, A. Basturk, S. Karacavus, and B. Yilmaz, “Metabolic Imaging Based Sub-Classification of Lung Cancer,” *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.3040155.

[4] Z. Lin *et al.*, “A unified matrix-based convolutional neural network for fine-grained image classification of wheat leaf diseases,” *IEEE Access*, vol. 7, 2019, doi: 10.1109/ACCESS.2019.2891739.

[5] S. Cebollada, L. Payá, M. Flores, A. Peidró, and O. Reinoso, “A state-of-the-art review on mobile robotics tasks using artificial intelligence and visual data,” *Expert Systems with Applications*, vol. 167. 2021. doi: 10.1016/j.eswa.2020.114195.

[6] M. Yousef, K. F. Hussain, and U. S. Mohammed, “Accurate, data-efficient, unconstrained text recognition with convolutional neural networks,” *Pattern Recognit*, vol. 108, 2020, doi: 10.1016/j.patcog.2020.107482.

[7] B. Epshtein, E. Ofek, and Y. Wexler, “Detecting text in natural scenes with stroke width transform,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010. doi: 10.1109/CVPR.2010.5540041.

[8] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, “Detecting texts of arbitrary orientations in natural images,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2012. doi: 10.1109/CVPR.2012.6247787.

[9] Y. Netzer and T. Wang, “Reading digits in natural images with unsupervised feature learning,” *Nips*, 2011.

[10] H. Badri, H. Yahia, and K. Daoudi, “Fast and accurate texture recognition with multilayer convolution and multifractal analysis,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2014. doi: 10.1007/978-3-319-10590-1_33.

[11] L. Neumann and J. Matas, “Scene text localization and recognition with oriented stroke detection,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2013. doi: 10.1109/ICCV.2013.19.

[12] S. Zhang, M. Lin, T. Chen, L. Jin, and L. Lin, “Character proposal network for robust text extraction,” in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2016. doi: 10.1109/ICASSP.2016.7472154.

[13] K. Wang, B. Babenko, and S. Belongie, “End-to-end scene text recognition,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2011. doi: 10.1109/ICCV.2011.6126402.

[14] C. Shi, C. Wang, B. Xiao, S. Gao, and J. Hu, “End-to-end scene text recognition using tree-structured models,” *Pattern Recognit*, vol. 47, no. 9, 2014, doi: 10.1016/j.patcog.2014.03.023.

[15] A. Mishra, K. Alahari, and C. V. Jawahar, “Top-down and bottom-up cues for scene text recognition,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2012. doi: 10.1109/CVPR.2012.6247990.

[16] C. Yao, X. Bai, B. Shi, and W. Liu, “Strokelets: A learned multi-scale representation for scene text recognition,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2014. doi: 10.1109/CVPR.2014.515.

[17] Y. Xin, D. Chen, C. Zeng, W. Zhang, Y. Wang, and R. C. C. Cheung, “High Throughput Hardware/Software Heterogeneous System for RRPN-Based Scene Text Detection,” *IEEE Transactions on Computers*, vol. 71, no. 7, 2022, doi: 10.1109/TC.2021.3092195.

[18] J. Ma *et al.*, “Arbitrary-oriented scene text detection via rotation proposals,” *IEEE Trans Multimedia*, vol. 20, no. 11, 2018, doi: 10.1109/TMM.2018.2818020.

[19] W. He, X. Y. Zhang, F. Yin, and C. L. Liu, "Deep Direct Regression for Multi-oriented Scene Text Detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017. doi: 10.1109/ICCV.2017.87.

[20] J. Memon, M. Sami, R. A. Khan, and M. Uddin, "Handwritten Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR)," *IEEE Access*, vol. 8. 2020. doi: 10.1109/ACCESS.2020.3012542.

[21] L. Q. Zuo, H. M. Sun, Q. C. Mao, R. Qi, and R. S. Jia, "Natural Scene Text Recognition Based on Encoder-Decoder Framework," *IEEE Access*, vol. 7, 2019, doi: 10.1109/ACCESS.2019.2916616.

[22] C. Yao, X. Bai, and W. Liu, "A unified framework for multioriented text detection and recognition," *IEEE Transactions on Image Processing*, vol. 23, no. 11, 2014, doi: 10.1109/TIP.2014.2353813.

[23] O. Y. Ling, L. B. Theng, A. Chai, and C. McCarthy, "A model for automatic recognition of vertical texts in natural scene images," in *Proceedings - 8th IEEE International Conference on Control System, Computing and Engineering, ICCSCE 2018*, 2019. doi: 10.1109/ICCSCE.2018.8685019.

[24] E. Zacharias, M. Teuchler, and B. Bernier, "Image Processing Based Scene-Text Detection and Recognition with Tesseract," Apr. 2020, Accessed: Mar. 31, 2023. [Online]. Available: https://keras-ocr.readthedocs.io/en/latest/

[25] "Zed2i Documentation." Accessed: May 01, 2023. [Online]. Available: https://www.stereolabs.com/docs/

[26] "Pixhawk Documentation." Accessed: May 01, 2023. [Online]. Available: https://pixhawk.org/

[27] O. M. T. Kaya and G. Erdemir, "Design of an Eight-Wheeled Mobile Delivery Robot and Its Climbing Simulations," in *Conference Proceedings - IEEE SOUTHEASTCON*, 2023. doi: 10.1109/SoutheastCon51012.2023.10115114.

[28] X. Zhou *et al.*, "EAST: An efficient and accurate scene text detector," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017. doi: 10.1109/CVPR.2017.283.

[29] Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee, "Character region awareness for text detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019. doi: 10.1109/CVPR.2019.00959.

[30] B. Shi, X. Bai, and C. Yao, "An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition," *IEEE Trans Pattern Anal Mach Intell*, vol. 39, no. 11, 2017, doi: 10.1109/TPAMI.2016.2646371.

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016. doi: 10.1109/CVPR.2016.90.

[32] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[33] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *ACM International Conference Proceeding Series*, 2006. doi: 10.1145/1143844.1143891.

[34] "Keras OCR Documentation." Accessed: Apr. 09, 2023. [Online]. Available: https://keras-ocr.readthedocs.io/

[35] R. Smith, "An overview of the tesseract OCR engine," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2007. doi: 10.1109/ICDAR.2007.4376991.

[36] "Difflib Library Documentation." Accessed: Apr. 09, 2023. [Online]. Available: https://docs.python.org/3/library/difflib.html

[37] J. W. Ratcliff and D. Metzener, "Pattern matching: The gestalt approach," *Dr. Dobb's Journal*, vol. 13, 1988.

BIOGRAPHIES

Erdal Alimovski received his B.Sc. degrees in Computer Engineering from St. Clement of Ohrid University-Bitola, North Macedonia in 2014, and the M.Sc. and Ph.D. degrees from Istanbul Sabahattin Zaim University, Istanbul, in 2020 and 2024, respectively. His research interests include computer vision, image processing, deep learning, machine learning, and robotics.

Gökhan Erdemir received the B.Sc., M.Sc., and Ph.D. degrees from Marmara University, Turkey. He was a Research Scholar with the Robotics and Automation Laboratory, Michigan State University, East Lansing, MI, USA, and the Health Management and Research Center, University of Michigan, Ann Arbor, MI, USA. He is currently an Associate Professor with the Engineering Management and Technology Department, The University of Tennessee at Chattanooga (UTC). His current research interests include control theory, robotics, industrial automation, AGVs, and engineering education.

Ahmet Emin Kuzucuoğlu received the B.Sc. degree from Electronics and Telecommunication Engineering Department, Istanbul Technical University, Turkey, in 1985, and the M.Sc. and Ph.D. degrees from Marmara University, Istanbul, Turkey, in 1994 and 2000, respectively. He is currently an Associate Professor with Marmara University. He has been in England and the United States of America in 1987 as part of YÖK-World Bank Vocational School Project. He has been in Lithuania in July-2006 as part of EU Leonardo da Vinci Type A mobility project. He is an Associate Professor with the Department of Electrical-Electronics Engineering. His current research interests include industrial automation, robotics, AI, control theory and applications.