CNN-Based Alphabet Identification and Sorting Robotic Arm

Saleh Al-Faraj, Mustafa Al-Bahrani, Saeed Al-Ghamdi, Marwan Rafie, Abul Bashar, and Ghazanfar Latif

Abstract Automated identification of objects and sorting them based on specified criteria is a crucial problem which is encountered by various manufacturing companies. To this end, they employ automated robots which need to perform these tasks with high accuracy. In this paper, a CNN-based machine learning system is proposed, designed, and implemented to accurately identify objects marked with English alphabets and sort them in a correct order based on the input given by the user. It consists of a hardware module which incorporates a robotic arm controlled by a Raspberry Pi microcontroller. The software module is based on a CNN-based image identifier model trained on an indigenous dataset consisting of 3898 images of English alphabets rotated at random angles. The experimental results demonstrate training and validation accuracies of 99.06% and 98.79%, respectively, based on the model trained over 120 epochs. Furthermore, our system was able to successfully sort and arrange the alphabets with the desired accuracy.

Keywords Alphabet sorting · Robotic arm · Image recognition · Convolutional neural networks · Robotic automation

S. Al-Faraj · M. Al-Bahrani · A. Bashar (✉) · G. Latif
Computer Engineering Department, Prince Mohammad Bin Fahd University, Khobar, Saudi Arabia
e-mail: abashar@pmu.edu.sa

S. Al-Faraj
e-mail: 201501220@pmu.edu.sa

M. Al-Bahrani
e-mail: 201601053@pmu.edu.sa

G. Latif
e-mail: glatif@pmu.edu.sa

S. Al-Ghamdi · M. Rafie
Computer Science Department, Prince Mohammad Bin Fahd University, Khobar, Saudi Arabia
e-mail: 201601001@pmu.edu.sa

M. Rafie
e-mail: 201600035@pmu.edu.sa

V. Bindhu et al. (eds.), *International Conference on Communication, Computing and Electronics Systems*, Lecture Notes in Electrical Engineering 733,
https://doi.org/10.1007/978-981-33-4909-4_26

1 Introduction

Generally speaking, sorting is such a common problem observed in various applications and researchers have tried to find fast and efficient ways to sort. Warehouse systems utilize the need for robots to manage their inventory and sort goods using automated systems. There have been many solutions toward automated sorting problem which will be presented in the next section. Alphabetical sorting is one problem that has not been given due attention and is a challenge the industry faces today. The main idea of this paper is to propose an automated robotic arm for identifying and sorting objects which have English alphabets written on them. Essentially, in our approach, a robotic arm is set up, which has the means to read images through a camera and identify the alphabet with the aid of computer vision algorithms. To make the system more interactive, the user inputs a word and based on this the robotic arm is supposed to pick the alphabets in the correct order and build the word with the correct spelling (i.e., sort the alphabets correctly). Robotic arm in general is one of the best options suitable for arranging and sorting problems. The proposed system utilizes a Raspberry Pi microcontroller to control the robotic arm movement, while the alphabet identification is performed with the help of a CNN-based machine learning model which is trained on our generated dataset.

The rest of this article is organized as follows. Section 2 discusses the background of this domain, while Sect. 3 presents the architectural details of the proposed system. Section 4 provides the results achieved and the related discussion. Section 5 concludes the paper with the main contributions and future recommendations.

2 Background

To design our system, the review of the literature is done for the automated sorting problem and found papers where the approach is either arranging objects or recognizing handwritten alphabets using a model which is pretrained dataset (such as the MNIST dataset) [1–4]. Digital image processing becomes important field with the emergence of new machine learning and deep learning techniques with the availability of high computational resources for the recognition, classification, and segmentation in different areas such as medical diagnosis from images [5], sign language recognition for disabled people [6, 7], traffic signs recognition [8], image enhancement [9, 10] and assisted living for visually impaired persons [11, 12]. Unable to find any research work that would arrange alphabets at a rotated angle which is scattered in a region of interest, it is believed that this is a new take on arranging scattered alphabets that are rotated at some random angle.

Table 1 provides a summary of the related work in this domain, where each approach is compared to our approach in terms of the key idea, their benefits, and drawbacks. This was also a discussion that wanted to know how beneficial our research work can be used as a real-world application. It is understood that some

Table 1 Comparison of different existing techniques with our proposed solution

References	Main idea	Benefits	Drawbacks
[13]	Recognizes alphabets based on their color. The arm detects the position of the alphabets from the image. The arm sorts the alphabets according to their color	Sorting colored objects from the panel and put them in the right boxes	There are many different items with the same color
[14]	Robotic arm sorts alphabets based on speech recognition. Alphabets are recognized based on RGB color and shape from the camera images	It avoids mistake and can carry heavyweight	The sound may face a problem since there are different accents diversity
[15]	A webcam is used to captures colored object cubes. Based on the color of the cubes, the robotics arm will place them into different cups	It works well for the manufacturer that uses color in organizing products	Based on the color of the image so leads to miss recognition if colors change
[16]	The camera image is processed using GNU Octave to determine the color and the shape	It is low cost and can sort colored objects	The objects around it can lead to wrong shape detection measurements
[17]	A handwritten optical character recognition (OCR) from the camera images to control the robotic arm	It can be used to scan handwritten notes and similar texts	Possibility for the wrong prediction if it is written by different persons
Proposed alphabet sorting robotic arm	The camera will take a picture of the workspace, and the alphabets are recognized using CNN that will send the position to the Robotics arm and arrange them accordingly	Using CNN as classifier gives high accuracy	It will only work for English Alphabets

industries have an arranging issue, where some packages would be labeled, and they would need to arrange them according to some criteria. One example application would be a library, as people come and borrow books. Once they return it, all these books would be added in some cart, shuffled, with labels that indicate which shelf should be placed alphabetically and it is a hassle for a librarian to sift through these books and find the "correct" book with the specified labeled to be added to the shelf. This is one possible application, however, there could be many other applications to use our proposed solution.

3 Proposed System Design

The proposed system has two major modules, namely the hardware and the software. The hardware module consists of the robotic arm, Raspberry Pi, and the servo controller. The software module is composed of the OCR system and the CNN-based image identifier and alphabet classifier. Their details are now presented below.

3.1 New Alphabets Dataset

A new data is prepared which has 3898 images A to Z capital alphabets as shown in Table 2. Firstly, the data processing packages were imported and then the images were captured from the images dataset. The sample of captured images is shown in Fig. 1.

Here, it can be seen that the alphabets are randomly rotated a certain angle. Next, they imported the machine learning packages, loaded the processed data, and split them into testing data and training data. The test data had 20% of the total number of images while the training data had 80% of the images. Later, a predefined model with four layers is created. The next step was to train the CNN model.

The model was then validated and found to be fit for testing. After that, came up with the trained and validate model which have been used to recognize the randomly arranged alphabets. The above-mentioned steps of alphabets image recognition are depicted in Fig. 2.

Table 2 Newly build alphabets dataset

A	190	F	128	K	155	P	127	U	131	Z	157
B	157	G	152	L	155	Q	153	V	149	Total	
C	158	H	155	M	155	R	137	W	152	3898	
D	126	I	156	N	157	S	140	X	149		
E	156	J	149	O	155	T	145	Y	154		

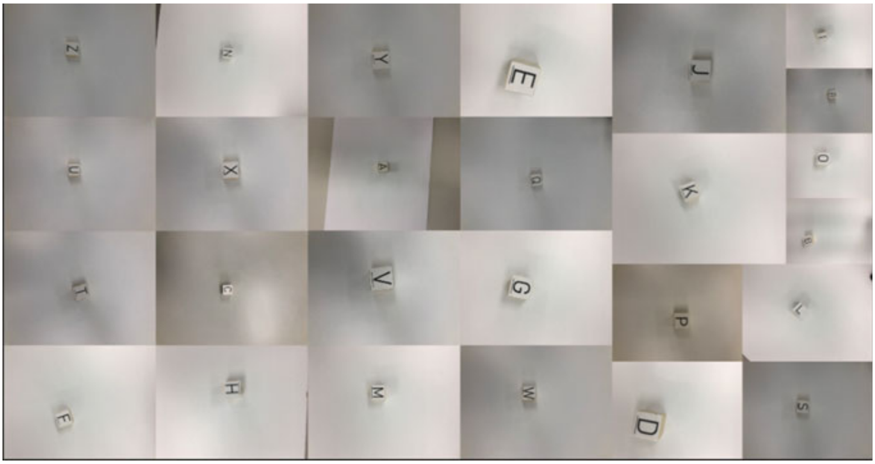

Fig. 1 Sample images of the newly built dataset

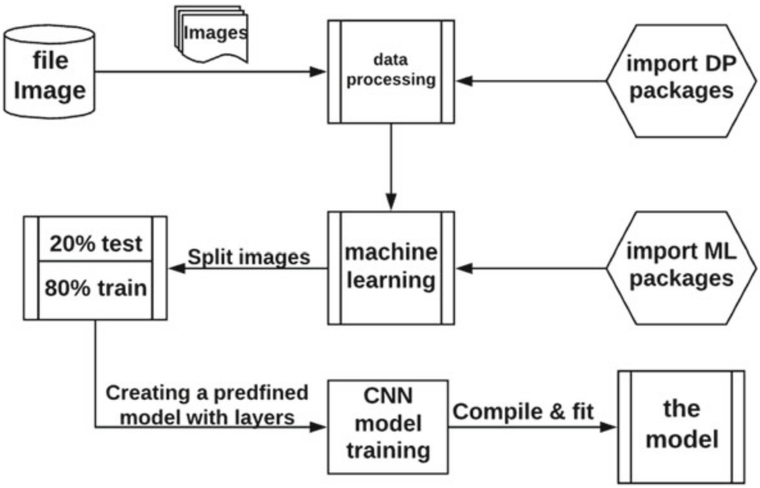

Fig. 2 Process of alphabets image recognition

3.2 Alphabets Image Processing

Now, the process of the alphabet OCR is described which is shown in Fig. 3. The camera takes the picture of the workspace which is saved as an image in a local file. Then, the alphabets in the image are recognized using optical character recognition (OCR). OCR is a technology that translates an image to be recognized by a machine. It distinguishes printed or handwritten text within digital images. Also, it is mostly used to scan documents. The software module first imports the packages and reads

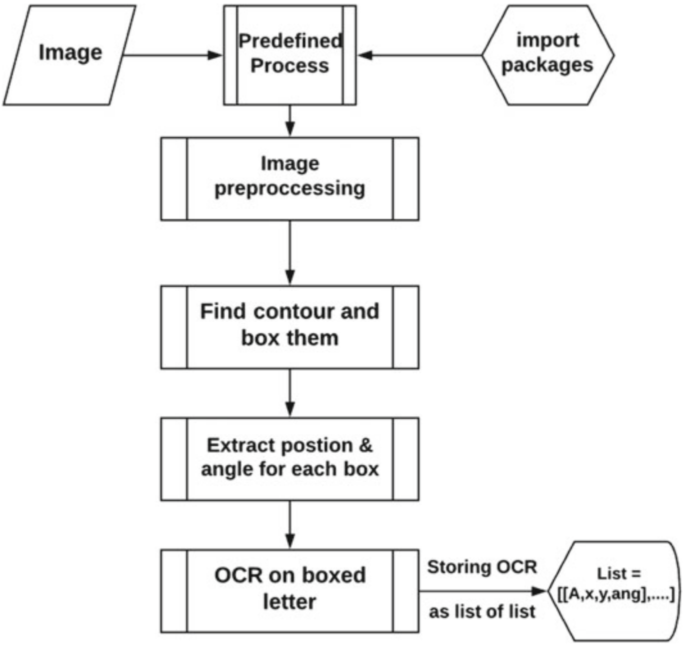

Fig. 3 Workflow of the proposed alphabets OCR

the input image. Then, it converts the image to grayscale and thresholds the image using the OpenCV library. After the recognition of the alphabet through CNN, the image is processed to find the contour and then they are boxed. After that, it extracts the position and the angle for each box. The midpoint for each box is then found, to have the arm fixed at the center of each box. Then in a repeated manner, it will perform the OCR on boxed alphabets to get the output as a list. The list will contain [*alphabets*, x and y coordinates, *rotation*]. The *alphabet* is one of the 26 possible English alphabets, the x and y coordinates represent the position on the workspace and the *rotation.* is the angle at which the alphabet is arranged. There are three main functions of the alphabets image processing, which are described below. They are preprocessing step, contour identification step, and the storage step.

Preprocessing: There is a need to preprocess the input which got as an image of the alphabets available on the workspace. The image is converted into black and white. Then threshold it using binary inversion. Also, there is a need to dilate the alphabets so that the OCR can better detect the desired alphabets.

Contour: After preprocessing our image, each alphabet has to be boded to figure out the rotated angle of each alphabet, locate the x-position and y-position of the alphabets, and perform an OCR to detect the alphabets shown in the box.

Data Storage: Once the OCR completes detecting each alphabet, it will store them as a list where each list contains four data variables: the alphabets detected, x-position, *y*-position and the angle. This array will then be passed to the robotic arm to loop through the alphabets and arrange them accordingly.

3.3 Alphabets Recognition Through CNN

Alphabets recognition was done with the convolutional neural network-based machine learning approach. Figure 4 shows the proposed CNN model used for image recognition where the image size is 48 × 48 and it passes through convolution 2D layer consisting of 48 filters with a kernel size of 3 × 3 and ReLU activation function [18, 19]. Then it passes through a max pooling 2D layer with 32 filters having a kernel size of 2 × 2 which returns the important features present in the image. This will result in the reduction of the image size. Then, it goes through another convolution layer of 24 filters with kernel size 3 × 3 and having a ReLU activation function. Later the image passes through a third convolution layer of 12 filters having a kernel size 3 × 3 and consisting of ReLU activation function. Then, having the classification layer that flattens the matrix and converts the vectors into a fully connected layer. Finally, the softmax function is used to classify the image into alphabets based on the predicted probability distribution value.

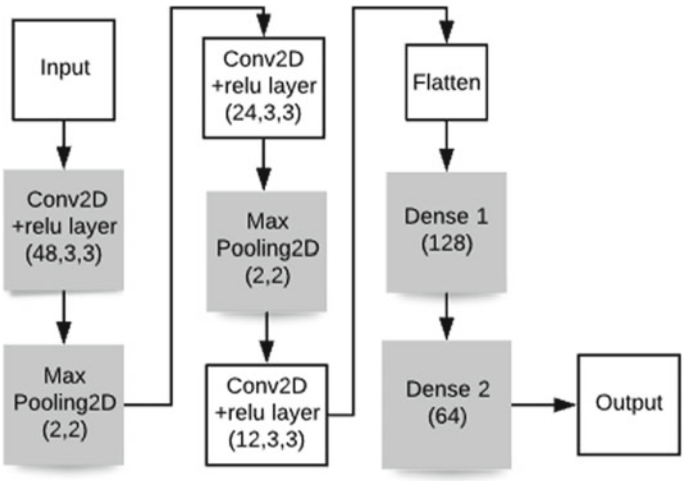

Fig. 4 Proposed CNN architecture

3.4 Hardware Design and Implementation

In the proposed system, different hardware components are integrated with the robotic arm which includes Raspberry Pi, digital camera, and RC servo motors. Figure 5 shows the circuit diagram of the hardware components. The hardware components are Raspberry Pi model B+, SSC-32U USB servo controller, six different types of HS servo motors, Pi camera, and two DC power supplies. The six HS servo motors are connected into six different channels in SSC-32U USB servo controller. Each channel contains three inputs which are pulse width modulation (PWM) pin, VCC pin, and ground pin. These inputs are arranged from top to bottom for each channel. Each servo motor has three outputs which are pulse width modulation (PWM) pin, VCC pin, and ground pin. SSC-32U USB servo controller requires a 12 V power supply connected to the VS2 because the channels that are used are connected to VS2. However, Raspberry Pi3 model B+ requires a 5 V power supply. The positive side of the DC power supply is connected to pin 4 in the Raspberry Pi, and the negative side of the DC power supply is connected to pin 6 in the Raspberry Pi. Raspberry Pi is connected with the SSC-32U USB servo controller through the USB cable. There is a special port in the Raspberry Pi (number 24) which is used to connect the Pi camera. The type of cable that is used to connect Raspberry Pi and Pi camera is CSI.

Robotic Arm: AL5D-PLTW arm is part of Lynxmotion's collection of AL5 robotic arm as shown in Fig. 6. This robotic arm has four degrees of freedom (4-DOF).

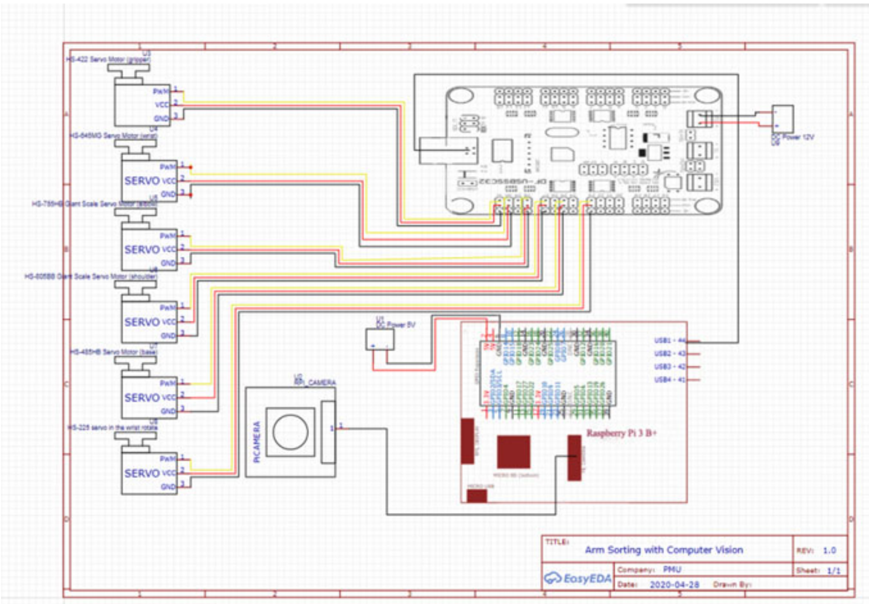

Fig. 5 Circuit diagram of the hardware components

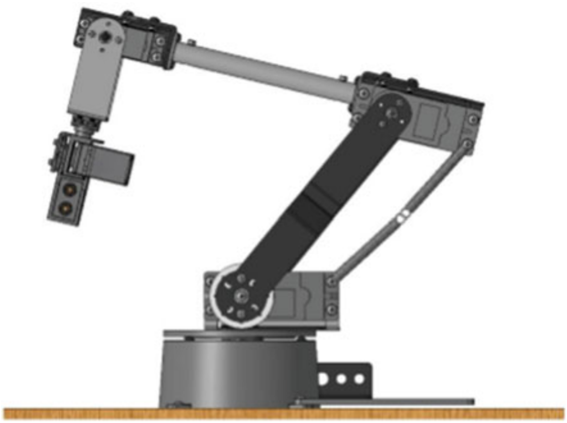

Fig. 6 AL5D robotic arm

The AL5D robotic arm can perform repeatable movements with high accuracy [20]. One important feature is that the arm can move at high speeds with precise positional placements [21]. Dimensions of the AL5D robotic arm: shoulder to elbow: 14.605 cm. Elbow to wrist: 18.7325 cm. Wrist to tip of gripper: 8.5725 cm. Height: 18.415 cm. Height (reaching up): 48.26 cm. Median forward reach: approx. 26.035 cm. Gripper opening: 3.175 cm. Weight: 0.878 kg. Range of motion per axis: 180°.

Servo Controller: SSC-32U is a dedicated servo controller, the core of R/C servo controller is an ATmega328p chip which has a Harvard architecture with an 8-bit RISC processor core as shown in Fig. 7 [22]. The servo controller was not supposed to be programmed but was meant to receive and execute commands sent to it from an external system such as a computer or microcontroller like Raspberry Pi. The R/C servo controller has many features including control up to 32 servo motors, USB, and serial input.

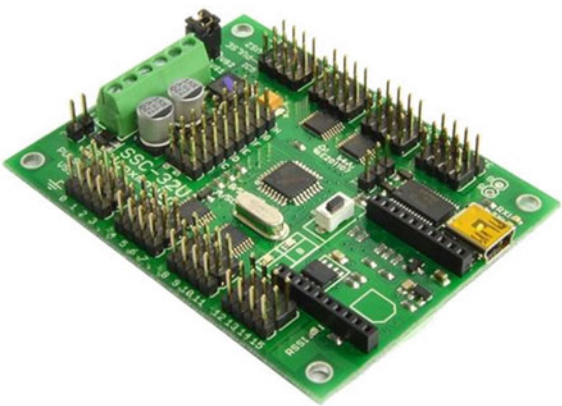

Fig. 7 SSC-32U servo controller

Raspberry Pi Microcontroller: It is a small credit card-sized computer. It is not only limited to perform routine home automation tasks, but can also be used for various other applications such as home entertainment, a video game console, or anything that is programmable [23]. Latest models of Raspberry Pi have in the better processing capabilities, new features, such as a wireless and Bluetooth chip. For our research work, it is found useful based on its relation to some of the system requirements desired, such as moving the robot arm to arrange the alphabets and executing a computer vision libraries which can recognize the alphabets. Therefore, the Raspberry Pi 3 B+ model is decided to use and installed the Raspbian OS to initiate our research work.

Workspace Setup: The workspace is made up of a square wooden board. It consists of two crossed planks, which are supported on top by three wooden planks as shown in Fig. 8. These crossed planks support the camera which is placed in the middle to capture the wooden board (workspace). The size of our wooden board is roughly around $20 \times 20\,\text{in}^2$. The workspace contains the alphabets which are scattered around having random angular rotations. The robotic arm is placed at the other end of the wooden board, so it will only be able to reach the alphabets of the workspace in front of it. The distance between the camera and the wooden board is around 64 in. The alphabets are written on cubic-shaped wooden boxes. Each side of the cubic wooden box is exactly 1 in., so the robotic arm grabber (end effector) will be able to grasp it and place it at the desired location on the workspace. The position information will be given as an input to the robotic arm from the output of the OCR module explained earlier.

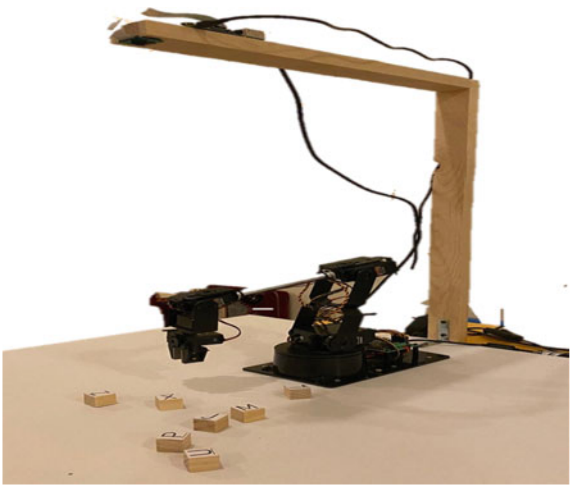

Fig. 8 Workspace setup for the robotic arm system

4 Experimental Results

In this section, the results are now presented from the experiments which were performed on the proposed system. Even though our system has two modules (hardware and software) focused on the results related to the software module, the hardware module was able to work properly based on the control signals provided to the Raspberry Pi controller from the CNN-based image classification module.

Two important accuracy measures which focused on were training accuracy and validation accuracy and also measured training loss and validation loss. These are important measures when evaluating the performance of CNN-based image classifier. The number of epochs was the independent variable which was used in this study. The number of epochs was varied from 1 to 120 and its effect on the four performance metrics is presented in Table 3. As it is expected, the accuracies increase with the number of epochs, whereas the loss functions decrease. Figure 9 shows that the training accuracy steadily increases from 21% to a maximum value of 99%, whereas

Table 3 Experimental results for alphabets recognition using CNN with different epochs

Epochs	Train Acc (%)	Train loss	Val. Acc (%)	Val. loss
1	21.46	2.7689	16.82	2.9277
15	65.83	1.0079	27.19	3.1293
30	77.81	0.6876	31.49	3.3836
60	87.50	0.4657	78.06	0.8680
120	99.06	0.2444	98.79	0.2408

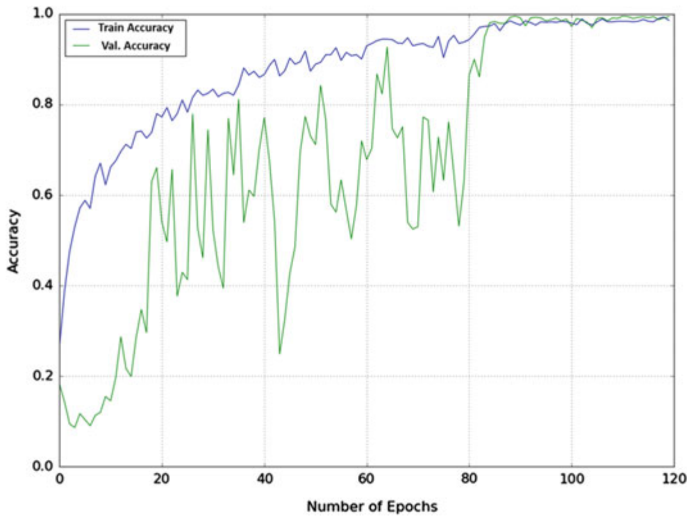

Fig. 9 Accuracy for alphabets recognition using CNN

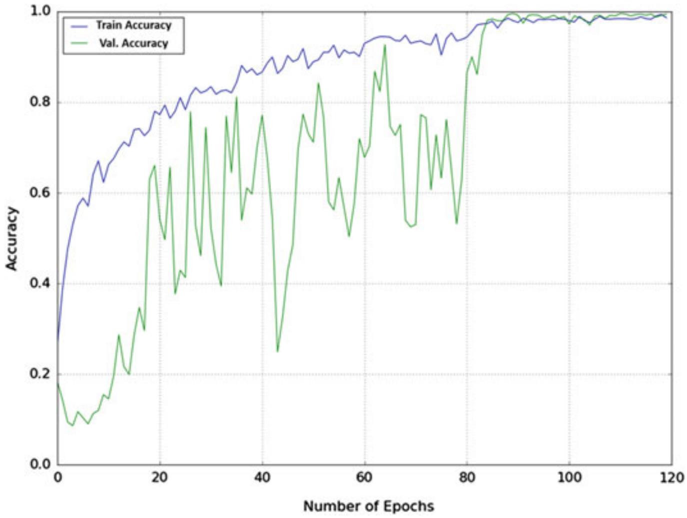

Fig. 10 Loss measurements for alphabets recognition

the validation accuracy initially has a fluctuated behavior and later on stabilizes to a maximum value of about 98.8%. Figure 10 provides a loss measurements for the recognition of the alphabet. As can be observed, both the training loss and the validation loss steadily decrease from a value of 2.77 to 2.93, respectively. At an epoch value of 120, the loss values reach a minimum value of 0.24 for both cases. By training the CNN model with 120 epochs, it found out that our system was able to correctly identify and classify the alphabets, as the accuracy was close to 99%. Further, it is observed that the robotic arm was able to correctly arrange the alphabets in the desired order according to the word which was given as an input to the system. Based on a variety of words, our system was able to demonstrate the achievement of the proposed objective which was English alphabet sorting.

5 Conclusions

In this paper, a deep learning-based CNN model has been proposed, designed, and implemented for recognizing and sorting English alphabets. Hardware and software modules are combined to provide a solution for arranging alphabets based on the desired word given as an input by the user. The hardware part of the system consists of a Raspberry Pi 3 Model B+, SSC-32U servo controller, and RC servo. The software part of the system was programmed in Python and using a convolutional neural network (CNN) machine learning model. The CNN model was trained on a dataset consisting of 3898 images of the 26 English alphabets randomly rotated at certain angles. The experimental results show that our model had a training and validation accuracies of 99.06% and 98.79%, respectively. As a part of future work, a system is planned to implement for Arabic alphabets. The changes that need to be implemented is the recognition of 28 Arabic alphabets and for this, the required layers to be used in the CNN model have to be studied. Another extension would be to speed up the training process by optimizing the various CNN layers and to reduce the identification time through faster processing microcontrollers.

Acknowledgements Our sincere thanks to the College of Computer Engineering and Science at Prince Mohammad Bin Fahd University for providing the resources and facilities for conducting this research.

References

1. Latif G, Alghazo J, Alzubaidi L, Naseer MM, Alghazo Y (2018) Deep convolutional neural network for recognition of unified multi-language handwritten numerals. In: 2018 IEEE 2nd ınternational workshop on Arabic and derived script analysis and recognition (ASAR). IEEE, pp 90–95
2. Al-Hmouz A, Latif G, Alghazo J, Al-Hmouz R (2020) Enhanced numeral recognition for handwritten multi-language numerals using fuzzy set-based decision mechanism. Int J Mach Learn Comput 10(1)
3. Alghazo JM, Latif G, Alzubaidi L, Elhassan A (2019) Multi-language handwritten digits recognition based on novel structural features. J Imaging Sci Technol 63(2):20502–20511
4. Alghazo JM, Latif G, Elhassan A, Alzubaidi L, Al-Hmouz A, Al-Hmouz R (2017) An online numeral recognition system using ımproved structural features–a unified method for handwritten Arabic and Persian numerals. J Telecommun Electron Comput Eng (JTEC) 9(2–10):33–40
5. Latif G, Iskandar DA, Alghazo JM, Mohammad N (2018) Enhanced MR image classification using hybrid statistical and wavelets features. IEEE Access 7:9634–9644
6. Latif G, Mohammad N, AlKhalaf R, AlKhalaf R, Alghazo J, Khan M (2020) An automatic Arabic sign language recognition system based on deep CNN: an assistive system for the deaf and hard of hearing. Int J Comput Dig Syst 9(4):715–724
7. Latif G, Mohammad N, Alghazo J, AlKhalaf R, AlKhalaf R (2019) ArASL: Arabic alphabets sign language dataset. Data Brief 23:103777
8. Alghmgham DA, Latif G, Alghazo J, Alzubaidi L (2019) Autonomous traffic sign (ATSR) detection and recognition using deep CNN. Procedia Comput Sci 163:266–274
9. Khan AH, Al-Asad JF, Latif G (2017) Speckle suppression in medical ultrasound images through Schur decomposition. IET Image Proc 12(3):307–313
10. Latif G, Iskandar DA, Alghazo J, Butt M, Khan AH (2018) Deep CNN based MR image denoising for tumor segmentation using watershed transform. Int J Eng Technol 7(2.3):37–42
11. AlSaid H, AlKhatib L, AlOraidh A, AlHaidar S, Bashar A (2019)Deep learning assisted smart glasses as educational aid for visually challenged students. In: Proceedings of ınternational conference on new trends in computing sciences (ICTCS 2019), Amman, Jordan, pp 1–6
12. AlZamil M, AlBugmi R, AlOtaibi S, AlAnazi G, AlZubaidi L, Bashar A (2020) COMPASS: IPS-based navigation system for visually ımpaired students. In: Proceedings of 9th IEEE ınternational conference on communication systems and network technologies (CSNT 2020), Gwalior, India, pp 161–166
13. Jia Y, Yang G, Saniie J (2017)Real-time color-based sorting robotic arm system. In: 2017 IEEE ınternational conference on electro ınformation technology (EIT), Lincoln, NE, pp 354–358. https://doi.org/10.1109/EIT.2017.8053385

14. AlSalman Z, AlSomali N, AlSayari S, Bashar A (2018) Speech driven robotic arm for sorting objects based on colors and shapes. In: 2018 3rd ınternational conference on ınventive computation technologies (ICICT), Coimbatore, India, pp 6–11. https://doi.org/10.1109/ICICT43934.2018.9034306
15. Szabó R, Lie I (2012)Automated colored object sorting application for robotic arms. In: 2012 10th ınternational symposium on electronics and telecommunications, Timisoara, pp 95–98. https://doi.org/10.1109/ISETC.2012.640811
16. Pereira V, Fernandes VA, Sequeira J (2014)Low cost object sorting robotic arm using Raspberry Pi. In: 2014 IEEE global humanitarian technology conference—South Asia Satellite (GHTC-SAS), Trivandrum, pp 1–6. https://doi.org/10.1109/GHTC-SAS.2014.6967550
17. Sarma P, Chourasia CK, Barman M (2019)Handwritten Assamese character recognition. In: 2019 IEEE 5th ınternational conference for convergence in technology (I2CT), Bombay, India, pp 1–6. https://doi.org/10.1109/I2CT45611.2019.9033603
18. Butt MM, Latif G, Iskandar DA, Alghazo J, Khan AH (2019) Multi-channel convolutions neural network based diabetic retinopathy detection from fundus images. Procedia Comput Sci 163:283–291
19. Shaikh E, Mohiuddin I, Manzoor A, Latif G, Mohammad N (2019) Automated grading for handwritten answer sheets using convolutional neural networks. In: 2019 2nd ınternational conference on new trends in computing sciences (ICTCS). IEEE, pp 1–6
20. LynxmotionLynxmotion SSC-32U USB Servo Controller Board. Retrieved from: https://www.robotshop.com/media/files/pdf2/lynxmotion_ssc-32u_usb_user_guide.pdf32u_usb_user_guide.pdf. Last accessed on 28 Apr 2019
21. Montiel-Vázquez EC, Torres-Rosique AE, Park G, Garcıa-Cavazos I, González-Flores U, y Mejia-Rosete RS, González-Hernández HG MOBMA: gesture driven mobile manipulator
22. Saleh T, Khan MR (2019) Hexapod robot for autonomous machining. In: IOP conference series: materials science and engineering, vol 488, no 1. IOP Publishing, p 012003
23. Agrawal N, Singhal S (2015) Smart drip irrigation system using raspberry pi and arduino. In: International conference on computing, communication and automation. IEEE, pp 928–932