

Rotten-Fruit-Sorting Robotic Arm: (Design of Low Complexity CNN for Embedded System) †

Muhammad Ihtisham Amin ¹, Muhammad Adeel Hafeez ^{2,*} and Qasim Awais ³

¹ Chenab College of Engineering and Technology, Gujranwala 52250, Pakistan; amin.ihtisham9@gmail.com

² Department of Electrical Engineering, Istanbul Medipol University, 34000 Istanbul, Turkey

³ Department of Electrical Engineering, GIFT University, Gujranwala 52250, Pakistan; qasim.awais@gmail.com

* Correspondence: muhammad.hafeez@std.medipol.edu.tr

† Presented at the 1st International Conference on Energy, Power and Environment, Gujrat, Pakistan, 11–12 November 2021.

Abstract: Industrial Automation has revolutionized the processing industry due to its high accuracy, the time it saves, and its ability to work without tiring. Being the most fundamental part of automation machines, robotic arms are being used as a fundamental component in many types of domestic as well as commercial automation units. In this paper, we proposed a low-complexity convolutional neural network (CNN) model and successfully deployed it on a locally generated robotic arm with the help of a Raspberry Pi 4 module. The designed robotic arm can detect, locate, and classify (based on fresh or rotten) between three species of Mangos (Ataulfo, Alphonso, and Keitt), on a conveyor belt. We generated a dataset of about 6000 images and trained a three-convolutional-layer-based CNN. Training and testing of the network were carried out with MatLab, and the weighted network was deployed to an embedded environment (Raspberry Pi 4 module) for real-time classification. We reported a classification accuracy of 98.08% in the detection of fresh mangos and 95.75% in the detection of rotten mangos. For the designed robotic art, the achieved angle accuracy was 93.94% with a minor error of only 2°. The proposed model can be deployed in many food- or object-sorting industries as an edge computing application of deep learning.



Citation: Amin, M.I.; Hafeez, M.A.; Awais, Q. Rotten-Fruit-Sorting Robotic Arm: (Design of Low Complexity CNN for Embedded System). *Eng. Proc.* **2021**, *12*, 109. <https://doi.org/10.3390/engproc2021012109>

Academic Editor: Shahid Iqbal

Published: 1 April 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: convolutional neural networks; edge computing; deep learning; fruit sorting; robotic arm

1. Introduction

Robots are an essential part of today's automation and processing industry and play their part in saving time and providing accuracy and ease. Robots have become a common companion in many applications from basic domestic robots, providing help in cleanliness and providing aid to disabled people, to high-accuracy assembling units in larger industries. As commercial companions, robots are proving to help in the loading and off-loading of objects, in picking and placing applications for conveyor belts, and in the computerized numerical control of different tools [1]. The conveyor-belt-based dragging and transportation of objects has become a fundamental part of the packing industry. In all of these robot applications, a robotic arm is always the most fundamental and valuable unit, which detects the object and picks and transports it to specified places. Traditional robotic arms were equipped with basic sensors, were operated via DC motors, and were only able to detect the presence of something on a conveyor and were unable to recognize them, especially when they had the same shape, weight, and dimensions [2,3]. Later on, with the advancement of image processing techniques, robotic arms became equipped with cameras and provided ease and accuracy for the sorting industry [4].

Over the past few years, there have been many inventions and a lot of progress in the field of AI. As a supervised learning approach, the CNN is a renowned approach to classify objects in images but is computationally expensive [5]. There has been a lot of work on the

hardware implementation of these computationally complex CCN-based classifiers. Many researchers have worked on and used different approaches for the deployment of CNNs in embedded platforms. In [6], the researcher proposed a Dian-Nao, which consumed a low amount of energy and had high throughput when it was processing a CNN. They have successfully optimized the memory impact on the moving design and the consumption of energy. In [7], optimized techniques were implemented for an accelerator-based CNN using the FPGA platform. In this study, Zhang et al. implemented and designed a lightweight CNN for embedded commodity hardware. In [8], a face detection system was designed based on a highly efficient CNN. In this project, Farabet et al. used Field Programmable Gate Array (FPGA) along with the external module of memory. In addition to these CNN-based research works, there are already many feature extractors such as shape, color, and size [9–12] that are specially designed for mobile platforms and embedded systems. In [9], the authors used the bottleneck technique to design a very small but highly effective network for classification. Forest et al. archived significantly high accuracy on ImageNet with much fewer parameters.

In this study, we created an application for an industrial robotic arm to sort, pick, and place fruits on a defined conveyer according to their physical condition (rotten or fresh). In the proposed prototype, the presence of an object on the production conveyer is sensed by an IR module, which sends the signal to the Raspberry Pi to stop the conveyer belt and capture the image using a camera. We used a simple RGB camera to capture images of three species of mangos on a conveyer and feed it to a pre-trained CNN, which was realized using a Raspberry Pi 4 embedded environment. We used a simple three-layer model of a CNN and trained it on over 6000 images, 2000 for each species, with 50% rotten mangos. To check the training accuracy, we used the test to train a ratio of 1:8. Once the recognition of mango is complete, the control unit sends the signal to the robotic arm to achieve associated angles, pick an object, and transfer it to a suitable supply conveyer.

2. Methodology

This study aimed to physically sort mangos in a packing industry to separate them based on their physical condition of being fresh or rotten. For this purpose, we captured images of mangos on the conveyer belt and fed them to a pre-trained CNN in an embedded system environment. We designed a low-complexity CNN which works on Raspberry Pi 4.

2.1. Data Acquisition

While collecting data on fruits, we chose three types of mangos (Ataulfo, Alphonso, and Keitt). For each type, we collected more than 1000 fresh mango pictures and 1000 rotten mango pictures, which means in total, we took 6000 pictures (3000 fresh and 3000 rotten). Samples of the data can be seen in Figures 1 and 2. The resolution of the camera we used was 2592×1944 pixels, which was connected by way of a 15 pin Ribbon Cable to a dedicated 15 pin Camera Serial Interface (CSI). For preprocessing, simple rescaling of the images was performed to $227 \times 227 \times 3$.



Figure 1. Fresh mangos.



Figure 2. Rotten mangos.

2.2. Classification Using CNN

Convolutional neural networks (CNNs) are widely used in different computer vision applications, especially in image recognition problems. This biologically inspired class of deep learning can map input images directly to the output classes. To classify our dataset of rotten and fresh mangos, we designed a CNN-based architecture, and the block diagram of the whole process, including training and recognition, is shown in Figure 3. The architecture was trained and tested on MATLAB before the prediction matrices were deployed to Raspberry Pi. We used arm compute library v20.05 to generate the code in C++ and Open CV to compile it on hardware. The CNN network used in the experiment had 3 convolution layers, each followed by normalization and ReLU. At the end of each convolution layer, we used max pooling of 3×3 to reduce the dimensionality. The first convolution layer had 36 filters of dimensions 7×7 . The second convolution layer had 72 filters of dimensions 3×3 , while the last convolution layer has 256 filters of dimensions 3×3 . At each convolution layer, we used the ‘same’ padding at each step. The last pooling layer was followed by a dropout layer with a ratio of 0.5, which was further connected to 2 fully connected layers. The last layer was SoftMax, which was used to compute the loss term and class probabilities in the classification. We did not use a pre-trained network; rather, we trained the network from scratch.

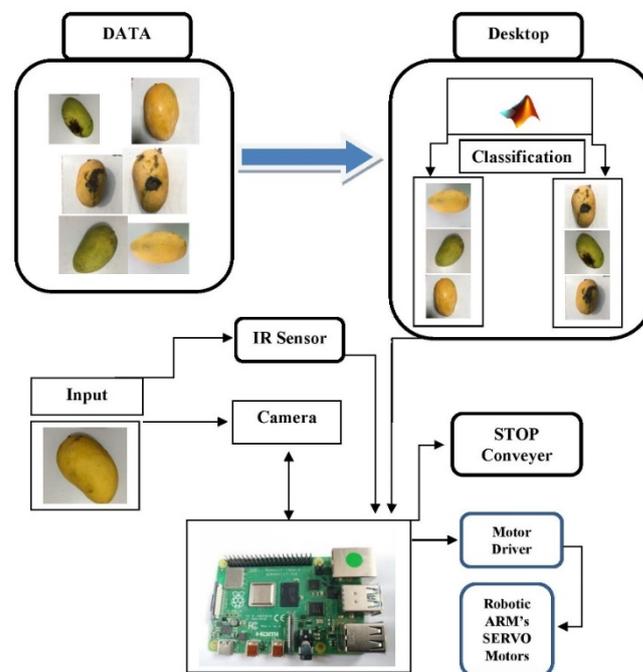


Figure 3. Block diagram indicating training to deployment process.

For training and testing purposes, we assigned the weights on a random basis. The stochastic gradient descent with momentum (SGDM) was used as an optimizer with a minimum batch size of 10 images per batch. In total, 30 epochs with initial learning rates of 0.01 were used for training, while the training to testing ratio was kept as 80/20.

2.3. Embedded System Implementation

This paper demonstrates a real-time classification application of a robotic arm in the fruit industry using Raspberry Pi 4 [13] and a camera module. The camera module was interfaced with Raspberry Pi 4, which extracted the dataset for classification, and later, it was also used to capture the images of mangos on the conveyer belt and forwarded them to Raspberry Pi 4. We used a CNN network developed in MATLAB as it was and deployed it over the Raspberry Pi using the ARM compute library [14] and the MATLAB support package for Raspberry Pi. To bring in the images from the camera port and to give instructions to servos, we used Open CV. In hardware, we used MG-90S for the elbow and shoulder of the robotic arm and SG-90 for the base and grip due to their light weight, stability, and accuracy in the given command and the achieved angle.

3. Results

The work proposed in this study was the implementation of a lower-complexity CNN on Raspberry Pi 4 modules to sort fresh and rotten mangos from the supply belt and to place them on a production belt. A three-layer CNN was designed with a classification accuracy of 98.08% in sorting fresh mangos, whereas the accuracy was 95.75% in the sorting of rotten mangos. The accuracy percentage of MG-90S was 93.94%, and SG-90 was 92.67%. The results show that there was less deviation from the commended angle in MG-90S because of its higher torque and accuracy. The embedded system performed with high accuracy. All of the parts functioned in a proper manner, and the response time was effective.

4. Conclusions

Convolutional neural networks are extensively used in different computer vision applications, especially for image recognition, but due to computational complexity, their embedded system implementation is rare. In this paper, we proposed a real-time application of a CNN in the sorting of rotten and fresh fruits with the help of a robotic arm. We designed a three-layer CNN architecture, which was highly accurate in recognizing mangos concerning their physical condition. The proposed CNN network was trained and tested on MATLAB, and then, the robotic arm carried out physical work using the Raspberry Pi 4 modules. We also made some design changes in our robotic arm to achieve an optimal system. The design prototype has many applications in different processes, such as in the food, packing, and automation industries. In the future, we would like to improve our system to achieve higher speeds in the physical sorting of objects and will enhance the application beyond the food industry.

Author Contributions: Conceptualization, data acquisition, methodology, software, validation, writing—original draft preparation, M.I.A., M.A.H.; writing—review and editing, M.A.H.; supervision, Q.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Dataset is available on request from corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Patil, R.B.; Kothavale, B.S.; Waghmode, L.Y.; Pecht, M. Life cycle cost analysis of a computerized numerical control machine tool: A case study from the Indian manufacturing industry. *J. Qual. Mainten. Eng.* **2020**, *27*, 107–128. [[CrossRef](#)]
2. Zheng, F.; Zecchin, A.C.; Newman, J.P.; Maier, H.R.; Dandy, G.C. An adaptive convergence-trajectory controlled ant colony optimization algorithm with application to water distribution system design problems. *IEEE Trans. Evol. Comput.* **2017**, *21*, 773–791. [[CrossRef](#)]
3. Qin, H.; Liu, M.; Wang, J.; Guo, Z.; Liu, J. Adaptive diagnosis of DC motors using R-WDCNN classifiers based on VMD-SVD. *Appl. Intell.* **2021**, *51*, 4888–4907. [[CrossRef](#)]

4. Font, D.; Palleja, T.; Tresanchez, M.; Runcan, D.; Moreno, J.; Martinez, D.; Teixido, M.; Palacin, J. A proposal for automatic fruit harvesting by combining a low-cost stereovision camera and a robotic arm. *Sensors* **2014**, *14*, 11557–11579. [CrossRef] [PubMed]
5. Alom, M.Z.; Hasan, M.; Yakopcic, C.; Taha, T.M.; Asari, V.K. Improved inception-residual convolutional neural network for object recognition. *Neural Comput. Appl.* **2020**, *32*, 279–293. [CrossRef]
6. Wilkinson, M.; Bell, M.C.; Morison, J.I.L. A Raspberry Pi based camera system and image processing-procedure for low cost and long-term monitoring of forest canopy dynamics. *Methods Ecol. Evol.* **2021**, *12*, 1316–1322. [CrossRef]
7. Kesav, N.; Jibukumar, M.G. Efficient and low complex architecture for detection and classification of Brain Tumor using RCNN with Two Channel CNN. *J. King Saud Univ.-Comput. Inform. Sci* **2021**, in press. [CrossRef]
8. Amin, M.I.; Hafeez, M.A.; Touseef, R.; Awais, Q. Person identification with masked face and thumb images under pandemic of covid-19. In Proceedings of the 2021 7th International Conference on Control, Instrumentation and Automation (ICCIA), Tabriz, Iran, 23–24 February 2021; pp. 1–4.
9. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50× fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
10. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
11. Guérin, J.; Thiery, S.; Nyiri, E.; Gibaru, O.; Boots, B. Combining pretrained CNN feature extractors to enhance clustering of complex natural images. *Neurocomputing* **2021**, *423*, 551–571. [CrossRef]
12. Howard, A.; Zhmoginov, A.; Chen, L.C.; Sandler, M.; Zhu, M. Inverted Residuals and Linear Bottlenecks: Mobile Networks for Classification, Detection and Segmentation. 2018. Available online: https://www.researchgate.net/publication/322517761_Inverted_Residuals_and_Linear_Bottlenecks_Mobile_Networks_forClassification_Detection_and_Segmentation (accessed on 16 July 2021).
13. Available online: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/> (accessed on 22 July 2021).
14. Mulfari, D.; Celesti, A.; Fazio, M.; Villari, M.; Puliafito, A. Using Google Cloud Vision in assistive technology scenarios. In Proceedings of the 2016 IEEE Symposium on Computers and Communication (ISCC), Messina, Italy, 27–30 June 2016; pp. 214–221.