Uluslararası İleri Doğa Bilimleri ve Mühendislik Araştırmaları Dergisi
Sayı 7, S. 70-76, 2, 2023

Araştırma Makalesi

IJANSER

https://as-proceeding.com/index.php/ijanser
ISSN: 2980-0811

International Journal of Advanced Natural Sciences and Engineering Researches
Volume 7, pp. 70-76, 2, 2023

Research Article

Real-Time Deep Learning based Tomato Fruit Quality Control in Conveyor Belt

Ali Furkan KAMANLI*[1]

[1]*Sakarya University of Applied Sciences, Faculty of Technology, Electrical and Electronics Engineering, Turkey*

(fkamanli@subu.edu.tr) Email of the corresponding author

(Received: 10 March 2023, Accepted: 26 March 2023)

(2nd International Conference on Scientific and Academic Research ICSAR 2023, March 14-16, 2023)

ATIF/REFERENCE: Kamanlı, A. F. (2023). Real-Time Deep Learning based Tomato Fruit Quality Control in Conveyor Belt. *International Journal of Advanced Natural Sciences and Engineering Researches*, 7(2), 70-76.

***Abstract* –**.Quality control is crucial in ensuring that the finished goods adhere to the specified requirements and standards in a manufacturing or production context. One key area where quality control is essential for sorting and separating goods on a conveyor belt. Automating this process can significantly improve the efficiency and accuracy of quality control. Manual quality control is a labor-intensive and time-consuming process that can lead to human error and inconsistencies. On the other hand, automation allows for the use of technology, such as machine vision and artificial intelligence, to quickly and accurately sort and separate goods based on predefined criteria. Automation also enables real-time monitoring and data collection, which can provide valuable insights into the manufacturing process and help identify areas for improvement.

Additionally, automation can improve the speed and efficiency of the sorting process, allowing for greater throughput of goods and increasing productivity. Furthermore, automation can reduce workplace injuries, as manual sorting and separation can be physically demanding and lead to repetitive strain injuries. In summary, automating quality control in the sorting and separating of goods on a conveyor belt is essential for improving the accuracy and efficiency of the process, reducing the risk of human error and workplace injuries, and providing valuable insights into the manufacturing process.

This paper used real-time deep learning-based tomato fruit quality control in conveyor belts with IP cameras. User interface and object detection models were used. For an automated separation procedure, tomato form, quality, size, and other variables were assessed in real-time. YoloV4 tiny, SSDMobileNet, and Faster RCNN models were utilized, and real-time accuracy of 88, 87, and 92 percent was attained. We carried out hyperparameter optimization, fine-tuning, and data augmentation.

Keywords – Deep Learning; Quality Control; Tomato; Real Time; Object Detection

I. INTRODUCTION

According to their size and quality, farmers typically segregate healthy and damaged tomatoes [1, 2]. The excellent quality of tomatoes makes it easier to distinguish between good and bad tomatoes and stops rotten tomatoes from spreading to quality tomatoes [3, 4]. Tomatoes that have been damaged are either thrown away or sold at a discount. Older individuals still sort tomatoes in the old-fashioned manner, which takes much time. Relying solely on a product flaw can result in a fundamental oversight when using human inspection. Customer feedback and happiness are significantly impacted by the desire to carry high-quality tomatoes with no discernible flaws. Image processing and machine learning techniques are used in this investigation.

Tomatoes and vegetables are among the most widely consumed food items in human life because millions of people use them daily. However, labor expenses have risen as the workforce ages, making many farms less economical. Tomatoes need to be produced more since, as was already noted, they are a widely used food item. Using a robot to harvest tomatoes rather than people is an efficient way to control quality and cut costs. As a result, most researchers have been developing fruit and vegetable harvesting robots during the past few decades [9, 10]. The primary indicator for ripening tomato detection is color. To be able to detect quality Qualityato fruit, various procedures are followed[11-12].

Utilizing IoT sensors to gather information on a product's quality during production is one strategy that could be implemented. IoT and camera-combined hybrid systems can be used to examine pictures or videos of the things being created using computer vision and deep learning [13-14]. This could entail spotting flaws or differences from the desired product's size or shape. As well as being used to discover patterns in the production process that could be exploited to increase efficiency, the data could also be used to analyze the manufacturing equipment performance [15-16].

In this study; deep learning models were used for real-time tomato quality control applications. An application deployed by a remote server will communicate with a high-quality IP camera that can be connected to the internet that will be placed at the quality control sites to take a real-time image of the production area. The application has created a system that continuously monitors the camera's position while it is operating, reads the incoming frames as input to the artificial intelligence model, and instructs the appropriate automation controller as it displays the output of the artificial intelligence model in the designed interface.

II. Materials and Method

The synchronized data flow between devices operating in a system designed to deliver a solution to a problem is the definition of IoT technology. Thanks to specific communication protocols, the system's devices can exchange data with one another. Google released the open-source communications technology known as GRPC (Google Remote Procedure Call).

In our study, identifying defects, measuring size and shape, or classifying tomato types is important. In our work, this is the first step to determine the type of data that needs to be collected and the metrics to evaluate the model's performance. The data needs to be diverse and representative of the different types of tomatoes and the different defects or variations that can occur. The data is labeled with the class or attribute of each tomato clearly defined. The data is prepared for use in the deep learning model by performing tasks such as resizing images, converting to grayscale, and removing unnecessary data. The training set is used to train the model. The validation set is used to tune the model's hyperparameters, and the test set is used to evaluate the final performance of the model. The test set is used to evaluate the model's performance to understand how well the model can detect or classify tomatoes and identify areas for improvement. Once the model's performance is suitable with the user interface, we deploy it to the inspection process, where the system can automatically classify or detect defects in real-time.

Most product quality checks in manufacturing facilities are performed on conveyors. Quality control workers can make mistakes even though products with undesired characteristics are identified and sorting procedures are carried out. Low-quality products can be shipped with other products due to human error. As a result, Figure 1 demonstrates the process flow and hierarchical structure of microservices interaction.

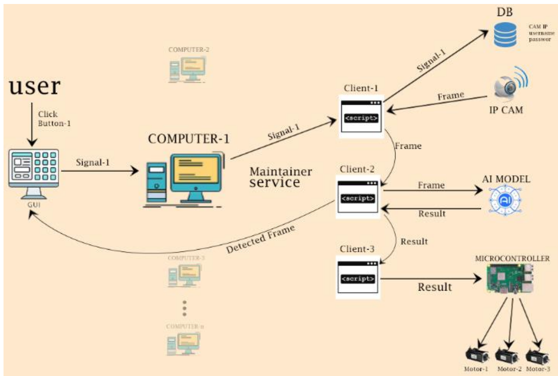

Figure 1. Remote-controlled tomato quality inspection electronic and database diagram

In a manufacturing or production setting, quality control is crucial to ensuring that the final products meet the desired specifications and standards. One key area where quality control is essential is sorting and separating goods on a conveyor belt. Automating this process can significantly improve the efficiency and accuracy of quality control.

Manual quality control is a labor-intensive and time-consuming process that can lead to human error and inconsistencies. On the other hand, automation allows for the use of technology, such as machine vision and artificial intelligence, to quickly and accurately sort and separate goods based on predefined criteria.

Automation also enables the use of real-time monitoring and data collection, which can provide valuable insights into the manufacturing process and help identify areas for improvement. Additionally, automation can improve the speed and efficiency of the sorting process, allowing for greater throughput of goods and increasing productivity.

Furthermore, automation can reduce workplace injuries, as manual sorting and separation can be physically demanding and lead to repetitive strain injuries.

In summary, automating the quality control in the sorting and separation of goods on a conveyor belt is essential for improving the accuracy and efficiency of the process, reducing the risk of human error and workplace injuries, and providing valuable insights into the manufacturing process.

Transfer learning is a technique in computer vision that allows a model trained on one task to be used for a different but related task. In the context of real-time quality control, transfer learning can be used to apply a model trained on a large dataset of available images to a smaller dataset of industrial images. This study evaluated YoloV4 tiny, SSDmobilenet, and Faster R-CNN models.

YOLOv4, SSDmobilenet, and Faster R-CNN are state-of-the-art real-time object detection models that can be fine-tuned to suit specific tasks through the transfer learning process. Transfer learning is a method of utilizing a pre-trained model and adapting it to a new task by fine-tuning it on a different dataset. In the case of deep learning, the pre-trained model has been trained on a large dataset of images and object classes and can be fine-tuned on a smaller dataset of images and object classes that are specific to the task at hand.

Hyperparameter optimization is a critical step in the process of fine-tuning a model. Hyperparameters are parameters that are not learned during the training process and include, for example, the learning rate and batch size. Fine-tuning a pre-trained model involves training the model on a new dataset while keeping some of the earlier layers fixed and re-training some of the last layers.

In real-world applications, transfer learning with Yolov4 tiny entails taking a pre-trained Yolov4 tiny model, optimizing a new dataset, and selecting the best hyperparameters for the new object detection. This is possible using methods like grid search or random search, which are used to identify the ideal set of hyperparameters.

Once the model has been fine-tuned, it can be used to make predictions on new images, and its performance can be evaluated using metrics such as mean average precision (MAP), as in our work.

Data augmentation is vital for deep learning object detection for several reasons:

Overfitting: Deep learning models can overfit if the dataset is small or needs to be more diverse. Data augmentation can increase the size and diversity of the dataset, making it more representative of the real-world scenarios that the model will encounter.

Robustness: Data augmentation can be used to simulate different variations of the objects being detected, such as different orientations, scales, and lighting conditions. This can improve the model's robustness to these variations, making it more likely to detect objects correctly in real-world scenarios.

Generalization: By training a model on a diverse set of augmented data, the model is exposed to different variations of the same object, allowing it to generalize better to new and unseen data.

Data Scarcity: Data augmentation can be a solution when there is not enough real-life data to train a deep learning model. Augmenting the data can increase the size of the dataset and improve the model's performance.

Cost: Collecting and labeling real-world data can be expensive and time-consuming. Data augmentation can be used to generate new data from existing data, allowing a model to be trained with fewer real-world examples.

Data augmentation is important for deep learning object detection because it can help increase the dataset's size and diversity, improve the model's robustness and generalization, and overcome the limitations of data scarcity and cost.

In our study, some examples of data augmentation techniques can be used for real-time tomato quality inspection using computer vision and deep learning.

- Rotation: Rotating the images of tomatoes by small angles can help the model learn to recognize tomatoes from different orientations.
- Translation: Shifting the images horizontally or vertically can help the model learn to recognize slightly off-center tomatoes.
- Scaling: Scaling the images up or down can help the model learn to recognize tomatoes of different sizes.
- Flipping: Flipping the images horizontally can help the model learn to recognize symmetrical tomatoes.
- Brightness and Contrast: Adjusting the brightness and contrast of the images can help the model learn to recognize tomatoes under different lighting conditions.
- Gaussian noise: Adding Gaussian noise to the images can help the model learn to recognize tomatoes affected by environmental conditions such as dust or moisture.
- Blur: Blurring the images can help the model learn to recognize tomatoes affected by motion blur.
- Crop: Cropping the images to focus on specific regions of the tomato, such as the stem or the blossom end, can help the model learn to recognize specific defects.

These are just a few examples of data augmentation techniques that are being used to improve the performance of a deep-learning model for real-time tomato quality inspection.

PyQt is a set of Python bindings for the Qt libraries that can be utilized to create graphical user interfaces (GUIs) for various applications. In this study, we demonstrate using PyQt in conjunction with the YOLOv4 object detection model.

The PyQt5 library must be installed and imported into the Python script. Next, a GUI application can be created that displays an image and runs the YOLOv4 object detection model on the image. The GUI should include elements such as a button to load an image, a button to initiate the object detection model, and a label to display the image with the detected objects.

In order to run the YOLOv4 object detection model, a pre-trained YOLOv4 model is loaded and utilized to make predictions on the image. Object detection in real time can be achieved using libraries such as OpenCV or TensorFlow. Additionally, the python library YOLOv4 can also be used to run the YOLOv4 model and obtain the detections in the image, which can then be displayed on the PyQt GUI.

Object detection models, such as YOLOv4, can be computationally demanding. Therefore, it may be necessary to optimize the application's performance by, for example, running the object detection model on a separate thread or using faster hardware, as shown in Figure 1.

In conclusion, this study demonstrates the use of PyQt combined with the YOLOv4 object detection model to create a user-friendly GUI that allows for intuitive loading and viewing images with detected objects.

Depending on the user's request and the message sent to the remote service, several stages can be employed in the user interface to raise a client. For instance, when clicked to the Connect-1 button, a unique ID is sent to the service, and a python script that will query the database is sent, which has been given a port number before it starts up. The user requests explicitly that this query run. Using a MySQL function, information for application-1 that is available in the database is obtained and returned as a return.

The information received includes details like the user name, password, and application name, as well as the IP address of the IP camera that has to be linked to the conveyor belt. In light of this knowledge, Client-1's responsibility is to access the camera, set up a loop, and capture images continually. Images from the camera are retrieved using the OpenCV library. Using the RTSP connection type, each frame is grabbed from the loop using the user name, camera IP, and password information.

III. Results and Discussion

In this study, YoloV4 tiny, Faster RCNN, and Mobilenet SSD models were used to detect the size and quality of tomatoes. The user interface connects conveyor belt systems, cameras, and embedded system computers. There are three different regions on the conveyor belt to detect the tomato quality and size.

Three quality metrics were evaluated (color, shape, and size) while the data labeling process. Trainings were compared for the same epoch number. The detection speed of models were also evaluated and shown in Table 1.

Object detection is a fundamental task in computer vision, which has a wide range of applications in various domains, such as autonomous vehicles, security systems, and robotics. The You Only Look Once (YOLO) algorithm is a real-time object detection framework, and YOLOv4 is the latest version of this algorithm. However, YOLOv4 may not be suitable for resource-constrained environments such as mobile and embedded devices. To address this issue, we propose using YOLOv4 Tiny TensorRT, a smaller and faster version of the model optimized for improved performance on NVIDIA GPUs.

TensorRT is a high-performance deep learning inference library developed by NVIDIA that can be used to optimize deep learning models for improved performance on NVIDIA GPUs. By using TensorRT to optimize YOLOv4 Tiny, we can expect faster and more efficient object detection in real-time applications.

In addition, TensorRT can also be used to optimize the performance of other deep learning-based computer vision models like Faster R-CNN. Faster R-CNN is an object detection algorithm that uses a region proposal network (RPN) to generate regions of interest (ROIs) in an image and a separate network to classify and refine the bounding boxes of the detected objects. TensorRT can accelerate the computation of the convolutional layers and reduce the precision of the model without sacrificing too much accuracy. TensorRT optimization can significantly speed up the object detection, especially running on NVIDIA GPUs. However, it is essential to note that TensorRT can only be used to optimize the model during inference rather than for training the model.

Moreover, deep learning-based image classification models can be used to automatically identify and classify the quality of tomatoes. These models can be trained to recognize various attributes such as the size, color, and shape of tomatoes and then use this information to classify them as high-quality or low-quality. TensorRT can be used to accelerate these models and thus improve the overall performance of the image classification process, resulting in faster and more accurate quality control. This can help to automate the quality control process in tomato production and increase the overall efficiency of the process.

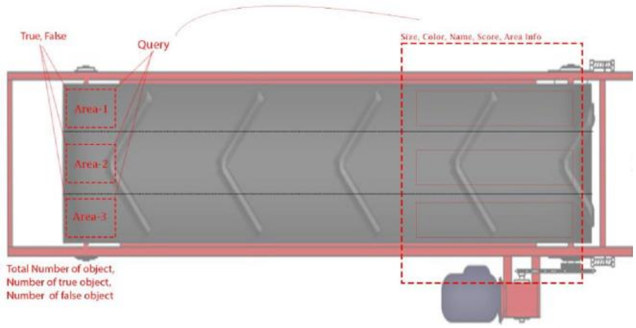

Figure 2. Conveyor belt visualization for progress.

Data augmentation was performed to evaluate the performance of object detection.

Table 1. Dataset size, after augmentation, five a fold of the original dataset

Model Data	Data Size
Training	8342
Test	2140
Validation	1210

As shown in Table 1, the preprocessed data were prepared for training in tomato quality inspection.

Table 2. Model accuracy table for training data and TensorRT (TRT) optimization

Model	**Accuracy**	**FPS**	**TRT**
Yolov4 Tiny	0.88	10	25
SSDMobileNet	0.87	12	18
Faster RCNN	0.91	2	-

The model has been trained and can be used to analyze images of tomatoes in a real-world setting. The model would output the location of each detected tomato in the image and a set of predictions for various quality metrics, such as size, color, and shape.

These predictions are compared against a set of predetermined standards for tomato quality, and any tomatoes that fall outside of these standards can be identified as being of poor quality.

Object detection models are used to effectively detect and classify tomatoes based on their quality by analyzing their visual features and comparing them against a set of predetermined standards. This process can be enhanced by using additional models trained explicitly for quality control. Figure 2 and Figure 3 demonstrate the conveyor belt region split for detection and pre-separation position.

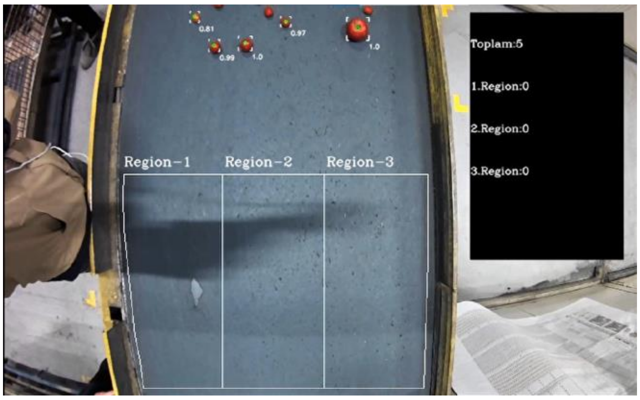

Figure 3. Conveyor belt usage for moving tomatoes.

Depending on the conveyor belt part, as shown in Figure 1, tomatoes go to the separate part for tomato delicacy. There are distinct picking places for tomatoes on the conveyor belt to prevent any damage while separating the quality. Figure 4 shows the real-time detection of the tomatoes in the conveyor belt.

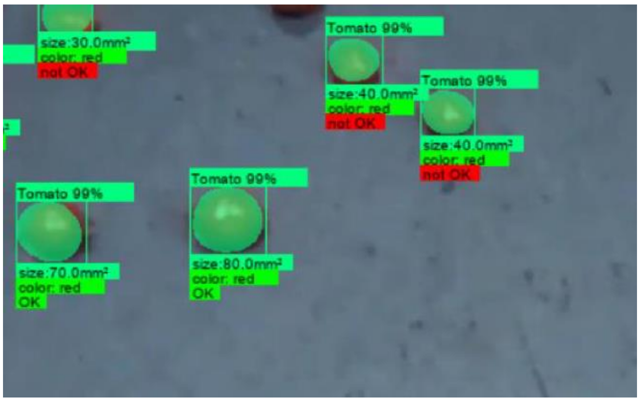

Figure 4. Tomato size and quality inspection in real-time example

After training, real-time experiments were carried out. Almost 50 thousand tomatoes were examined, and overall accuracy was %93 and % 91 in the Yolov4 tiny and SSDMobileNet models, which can show the system's actual performance rather than the model's training accuracy. Faster RCNN model accuracy score was higher, but due to real-time optimization problems, the model cannot be implemented to a conveyor belt for real-time accuracy due to its nature.

IV. CONCLUSION

Tomato quality is essential for packaging because it directly impacts the product's appearance, shelf life, and overall consumer satisfaction.

In terms of appearance, high-quality tomatoes are typically more visually appealing, with bright, consistent color and smooth skin. This can be more attractive to consumers, which can lead to higher sales and a better reputation for the product.

In terms of shelf life, high-quality tomatoes typically have a longer shelf life, which means that they can be stored for a more extended period of time before being sold. This can be beneficial for both retailers and consumers, as it allows for a longer window of time for the tomatoes to be sold and for consumers to use them before they go bad.

Finally, high-quality tomatoes are likely to be more flavorful, which can impact the consumers' overall satisfaction with the product. Consumers who have a positive experience with a product are more likely to purchase it again in the future, which can lead to repeat sales and customer loyalty.

Quality control of tomatoes at the packaging stage ensures that only the best tomatoes are being packaged, which can lead to better sales and customer satisfaction. The process also helps minimize product loss due to defects and damage.

ACKNOWLEDGMENT

Sakarya University of Applied sciences BAP supports this study with project number 078-2022.

REFERENCES

[1] Pogonyi, A., Pék, Z., Helyes, L., & Lugasi, A. (2005). Effect of grafting on the tomato's yield, quality, and main fruit components in spring force. Acta Alimentaria, 34(4), 453-462.

[2] A. Mohapatra, S. Shanmugasundaram, and R. Malmathanraj, ‘‘Grading of ripening stages of red banana using dielectric properties changes and image processing approach,’’ Comput. Electron. Agricult., vol. 143, no. 382,
pp. 100–110, 2017.

[3] De Luna, R. G., Dadios, E. P., Bandala, A. A., & Vicerra, R. R. P. (2019). Tomato fruit image dataset for deep transfer learning-based defect detection. In 2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation, and Mechatronics (RAM) (pp. 356-361). IEEE.

[4] Sanaeifar, A., ZakiDizaji, H., Jafari, A., & de la Guardia, M. (2017). Early detection of contamination and defect in foodstuffs by electronic nose: A review. TrAC Trends in Analytical Chemistry, 97, 257- 271.

[5] da Costa, A. Z., Figueroa, H. E., & Fracarolli, J. A. (2020). Computer vision based detection of external defects on tomatoes using deep learning. Biosystems Engineering, 190, 131-144.

[6] Zhang, B., Huang, W., Li, J., Zhao, C., Fan, S., Wu, J., & Liu, C. (2014). A review of the principles, developments and applications of computer vision for external quality inspection of fruits and vegetables. Food Research International, 62, 326-343.

[7] Blasco, J., Aleixos, N., Molto, E., 2007. Computer vision detection of peel defects in citrus by
means of a region oriented segmentation algorithm. J. Food Eng. 81 (3), 535–543.

[8] H. Lu, F. Wang, X. Liu, and Y. Wu, ‘‘Rapid assessment of tomato ripeness using visible/near-infrared spectroscopy and machine vision,’’ Food Anal. Methods, vol. 10, no. 6, pp. 1721–1726, 2017.

[9] S. Bargoti and J. Underwood, ‘‘Deep fruit detection in orchards,’’ in Proc. IEEE Int. Conf. Robot. Autom., May 2017, pp. 3626–3633

[10] Sembiring, A., Budiman, A., & Lestari, Y. D. (2017). Design and control of agricultural robot for tomato plants treatment and harvesting. In Journal of Physics: Conference Series (Vol. 930, No. 1, p. 012019). IOP Publishing.

[11] Oktarina, Y., Dewi, T., Risma, P., & Nawawi, M. (2020). Tomato harvesting arm robot manipulator; a pilot project. In Journal of Physics: Conference Series (Vol. 1500, No. 1, p. 012003). IOP Publishing.

[12] Lili, W., Bo, Z., Jinwei, F., Xiaoan, H., Shu, W., Yashuo, L., ... & Chongfeng, W. (2017). Development of a tomato harvesting robot used in greenhouse. International Journal of Agricultural and Biological Engineering, 10(4), 140-149.

[13] Feng, Q., Zou, W., Fan, P., Zhang, C., & Wang, X. (2018). Design and test of robotic harvesting system for cherry tomato. International Journal of Agricultural and Biological Engineering, 11(1), 96-100.

[14] Jahns, G., Nielsen, H.M., Paul, W., 2001. Measuring image analysis attributes and modelling fuzzy consumer aspects for tomato quality grading. Comput. Electron. Agric.31(1),17–29.https://doi.org/10.1016/S0168-1699(00)00171-X

[15] Moallem, P., Serajoddin, A., Pourghassem, H., 2017. Computer vision-based apple grading for golden delicious apples based on surface features. Inf. Process. Agric. 4 (1), 33–40.

[16] [16] Moltó, E., Blasco, J., Gómez-Sanchís, J., 2010. Analysis of hyperspectral images of citrus fruits. Hyperspectral Imaging for Food Quality Analysis and Control. Elsevier, pp. 321–348