JRAEE

Journal of Robotics, Automation, and Electronics Engineering
Vol. 4, No. 1, 2026, pp. 237-249
e-ISSN 3025-3780
https://journal.uny.ac.id/v3/jraee

Prototype of a Coffee Bean Weight Measuring Device Using a Webcam with the Convolutional Neural Network (CNN) Method at Roetin Coffee Shop

Nurul Afifah[a,1], Faris Yusuf Baktiar[b,2,*]

[a,b]Department of Electrical and Electronics Engineering, Vocational Faculty, UNY
[1]nurulafifah.2021@student.uny.ac.id; [2]farisyusufbaktiar@uny.ac.id
[*]Corresponding Author

ARTICLE INFO

Article History
Received Date Month Year
Revised Date Month Year
Accepted Date Month Year

ABSTRACT

The advancement of Artificial Intelligence (AI) and Computer Vision has enabled new opportunities for automation within the coffee industry, particularly in weight measurement of coffee beans, which is still performed manually and becomes inefficient at large scale. This study proposes an automatic weight estimation system using images captured by a webcam and processed through a Convolutional Neural Network (CNN) employing MobileNet as a lightweight regression model. The developed system analyzes visual features to estimate weight autonomously, offering an efficient, contactless alternative to conventional weighing tools and supporting stock monitoring for coffee industries and small enterprises.

ABSTRAK

Perkembangan kecerdasan buatan (AI) dan pengolahan citra digital (Computer Vision) membuka peluang baru dalam otomasi industri kopi, termasuk proses pengukuran berat biji kopi yang selama ini masih dilakukan secara manual dan kurang efisien dalam skala besar. Penelitian ini merancang sistem estimasi berat biji kopi dalam toples transparan menggunakan citra dari webcam yang diproses dengan Convolutional Neural Network (CNN) arsitektur MobileNet sebagai model regresi untuk memprediksi berat berdasarkan fitur visual. Sistem yang dikembangkan bersifat otomatis dan berpotensi menjadi alternatif timbangan konvensional serta mendukung monitoring stok pada industri kopi maupun UMKM.

Keywords
CNN;
Image Processing;
Mobile Net;
Coffee Beans.

1. Introduction

The coffee shop industry in Indonesia is experiencing rapid growth, making raw material inventory management a vital component for maintaining production continuity [1, 2]. However, Roetin Coffee Shop still performs stock recording manually using paper and packaging-based quantity estimation rather than actual weight, which increases the risk of data discrepancies and record loss, especially since the remaining coffee beans inside the grinder are not reweighed [3, 4]. Inaccurate stock opname may lead to shortages of essential raw materials that disrupt sales activities [5], indicating the need for an automatic inventory system

 ISSN 3025-3780 (online)
ISSN 3025-4590 (printed)

that is more accurate, efficient, and real-time [6, 7].

Advancements in Artificial Intelligence and Computer Vision enable the implementation of image based weight estimation using Convolutional Neural Networks (CNN), where MobileNet is selected for its efficiency and speed in visual feature extraction [8, 9, 10]. Similar studies have been conducted on palm fruit weight prediction [11], cattle weight estimation using image processing [12], digital inventory and stock opname systems [5], and load-cell-based measurement approaches [13, 4]; however, no studies have been found that apply CNN specifically for direct coffee bean weight estimation to support coffee shop inventory management. Therefore, this research proposes an automatic coffee bean weight estimation system using a webcam and MobileNet, with its main contribution being the integration of weight measurement and real time inventory logging to improve stock accuracy [14].

2. Problem Solving Approach

2.1 Computer Vision

In this study, a Computer Vision approach was selected as the main method for estimating the weight of coffee beans. This approach was chosen because it can capture visual features of objects directly from images, without requiring physical contact like manual weighing. The visual characteristics of coffee beans, including size, color, surface texture, and the distribution of beans within the container, can be digitally processed and analyzed as indicators of weight estimation. This method also allows measurements to be conducted quickly, continuously, and automatically using only a cam-era, making it highly suitable for monitoring daily stock at Roetin Coffee Shop, which requires time and labor efficiency. Therefore, Computer Vision becomes an effective and practical alternative to conventional weighing methods that still require manual interaction and are prone to recording errors.

2.2 Convolutional Neural Network (CNN)

After determining the visual approach, the classification and feature extraction model used is the Convolutional Neural Network (CNN). CNN was chosen due to its superior capability in recognizing visual patterns compared to traditional machine learning methods, which still rely on manual feature engineering. CNN can automatically extract important information from images through convolutional layers, so details such as bean shape, pile density, and color intensity can be accurately analyzed. In the context of this study, the ability of CNN to map the visual relationship between coffee bean images and weight values is essential to achieve precise estimation results. With high efficiency and accuracy, CNN provides the best foundation for building an image-based coffee bean weight prediction model.

2.3 MobileNet (CNN Architecture)

To ensure the system can run lightweight and fast on devices with limited computational resources, such as operational laptops and webcams, the MobileNet architecture was chosen as the main CNN model.

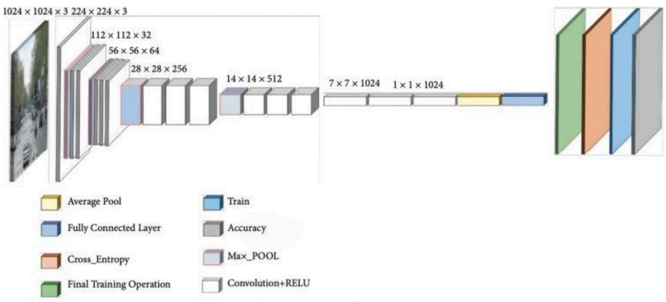


Fig. 1 Architecture Mobile Net

MobileNet is specifically designed to be more efficient in memory and computational usage through the depthwise separable convolution mechanism, allowing training and prediction processes to run much faster compared to larger CNN architectures, as illustrated in Fig. 1. This advantage makes MobileNet not only efficient but also ideal for real-time applications in work environments such as coffee shops, which require quick response in stock recording. With the selection of MobileNet, the coffee bean weight estimation system can operate optimally without high-specification devices, thereby increasing the potential for direct implementation in business operations.

3. Research Method

This study uses the Research and Development (R&D) method, which includes stages from system design to testing to evaluate the performance of a prototype automatic coffee bean weight measurement device, as shown in Fig. 2. The system is developed using image processing technology based on MobileNet integrated with a webcam, so that coffee bean weight measurement is no longer performed manually using a scale, but automatically through image analysis. The research stages include problem identification related to manual stock recording prone to errors, literature review and collection of a coffee bean image dataset in various containers and fill levels, design of the device including the placement of main components such as the Raspberry Pi and OLED display, system implementation with training of the CNN MobileNet model, and prototype testing and evaluation based on the accuracy of weight estimation compared to a digital scale.

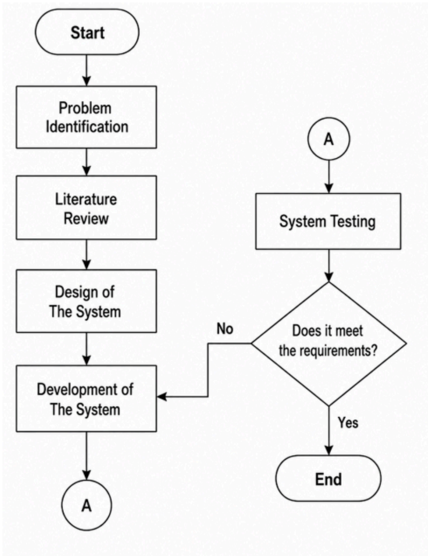


Fig. 2 Research method

The research was conducted at Roetin Coffee Shop, Trimulyo I, Kepek, Kec. Wonosari, Kab. Gunungkidul, Yogyakarta, from April 2025 to November 2025. The success of the system is measured based on its ability to produce consistent weight estimates with a low error margin (less than 5%) and the ease of monitoring daily coffee bean stock automatically through an application interface that displays real time results.

3.1 Requirements Analysis

The automatic coffee bean weight measurement system requires hardware that supports image acquisition, data processing, and real-time result display. A laptop is used for storing and managing the coffee bean image dataset as well as for visual verification before the model is trained, while the Raspberry Pi 4 Model B serves as the main processing unit, handling all image processing, MobileNet model inference, and displaying the estimated weight results. A webcam captures images of the coffee jars automatically, a tripod stabilizes the camera to ensure consistent images, and a 1.3-inch OLED screen displays the estimated results with high contrast and good visibility, even under low-light conditions. The complete setup of the required hardware can be seen in Fig. 3, which illustrates all the main components used in this system.

Fig. 3 Hardware requirements

The software used includes Visual Studio Code as the source code editor for developing Python programs, including image processing, training and implementing the MobileNet model, and creating a simple user interface. Python is the main programming language due to its flexibility in executing the entire system logic, from image acquisition via webcam, image pre-processing

ISSN 3025-3780 (online)
ISSN 3025-4590 (printed)

using OpenCV, to training and inference of the MobileNet model for automatic coffee bean weight estimation.

3.2 System Architecture Design

The proposed system for automatic coffee bean weight measurement utilizes a visual approach based on a webcam and an artificial intelligence model, which can be seen in the system architecture shown in Fig. 4. In this system, the webcam captures daily images of jars containing coffee beans. The images are then processed on the server using the OpenCV library for image pre-processing, including resizing, contrast adjustment, and segmentation of the coffee bean area within the jars. After pre-processing, the images are input into a pre-trained MobileNet model to estimate the coffee bean weight based on visual characteristics such as height, density, and color of the jar contents. The model runs on a Raspberry Pi and outputs weight estimations in grams without the need for physical sensors like load cells. The estimated data can be displayed on a website for daily stock recording and stored in a database for future reference, as shown in Fig. 4.

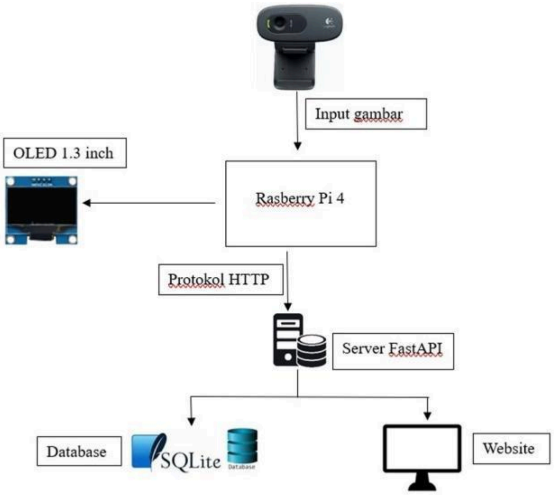


Fig. 4 System architecture design

To support model training, a dataset was collected to provide both training and testing data for the MobileNet-SSD model. The dataset consists of seven object classes: toples1, toples2, toples3, toples4, plastik1kg, plastik2kg, and bijikopi. Images were captured from multiple angles and lighting conditions to increase dataset variability. All collected images were annotated manually using Labeling software, as can be seen in Fig. 5, in Pascal VOC (XML) format to ensure accurate bounding box positioning for each object. The annotated dataset was divided into training (70%), validation (20%), and test (10%) subsets, with 2,694 images for training, 769 for validation, and 386 for testing. The training set enables the model to learn object patterns, the validation set evaluates model performance during training to prevent over-fitting, and the test set assesses the model's generalization capability on previously unseen data.

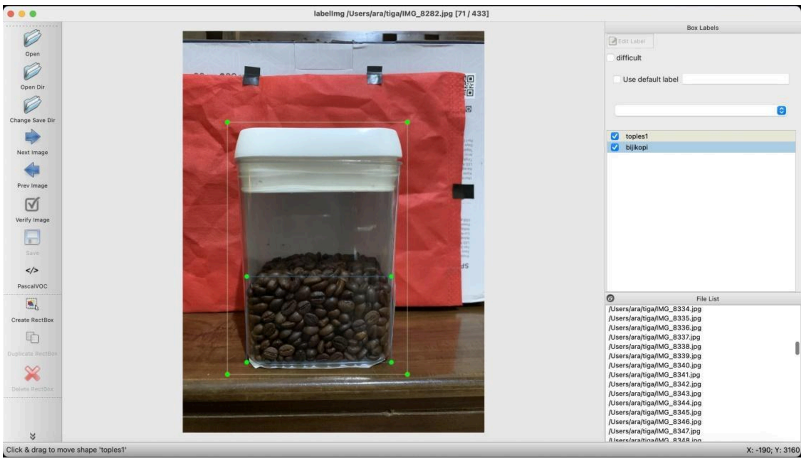


Fig. 5 Labelling system

3.3 Image Processing Algorithm Flow for Coffee Beans

The image processing in the coffee bean weight estimation system based on computer vision using MobileNet consists of four main stages: image acquisition, pre-processing, feature extraction, and weight prediction.

3.3.1 Image Acquisition

Images were captured using a Logitech C270 camera connected to a Raspberry Pi. The camera acquired frontal images of transparent jars containing varying amounts of coffee beans, as shown in Fig. 6 and Fig. 7, with the jars positioned perpendicular to the surface. The lighting was adjusted to minimize reflections on the jar surfaces that could affect image quality. Captured images were automatically stored in the dataset folder according to their actual weight labels, with 5–10 images per category used for training.

3.3.2 Image Pre-Processing

The image pre-processing stage consists of several key steps. First, the images are resized to 320×320 pixels to match the input layer of the model and reduce computational load. Next, normalization is applied, converting pixel values from the range 0–255 to 0–1 to stabilize the distribution of values and improve model performance. During training, data augmentation such as rotation, horizon-tal flipping, and brightness adjustment is performed to increase the model's robustness to variations in lighting and camera angles. Additionally, the image color format is converted from BGR (Blue, Green, Red) to RGB (Red, Green, Blue) so that the colors interpreted by the model accurately reflect the original image.

Fig. 6 Image acquisition

Fig. 7 Front view image capture

3.3.3 Feature Extraction

Feature extraction is carried out by the MobileNet architecture integrated with a Single Shot Detector (SSD). MobileNet extracts visual patterns from RGB images, from low-level features such as edges and textures to high-level features such as object shapes and characteristics. These features are processed by a Feature Pyramid Network (FPN) to ensure that both small and large objects are consistently recognized. The extracted features are then used by the SSD to locate objects (bounding boxes) and classify them into categories such as jars, plastic bags, and coffee beans.

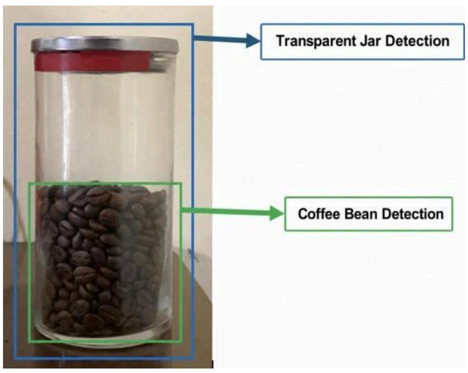


Fig. 8 Detection of transparent jars and coffee beans

The identification of important areas within the image is illustrated in Figure 8, where the trans-parent jar area is highlighted with a blue box, and the coffee bean contents are marked with a green box.

Journal of Robotics, Automation, and Electronics Engineering
Vol. 4, No. 1, 2026, pp. 237-249
ISSN 3025-3780 (online)
ISSN 3025-4590 (printed)

3.3.4 Coffee Bean Weight Prediction

Each type of container has a unique shape and volume, meaning the relationship between the height of the coffee bean content (cm) and its weight (grams) varies. Weight estimation is performed separately for each container using a linear approach based on empirical calibration. The height-to-weight relationship is calculated using linear interpolation as follows:

$$w = w_1 + (w_2 - w_1) \times \frac{(h - h_1)}{(h_2 - h_1)} \tag{1}$$

Where:
- w = estimated weight of coffee beans (grams)
- h = detected height of coffee beans (cm)
- h_1, h_2 = reference heights from the calibration table (cm)
- w_1, w_2 = reference weights from the calibration table (grams)

3.4 Hardware Design

The hardware design of the coffee bean weight estimation system using visual images consists of five main components Raspberry Pi 4 Model B, 1.3-inch OLED, laptop, webcam, and tripod. The laptop serves as the central hub for dataset management and program development, including image acquisition from the webcam, image pre-processing using OpenCV, and training the MobileNet model. Once the training is complete, the model is deployed on the Raspberry Pi, which acts as the main computational unit to process input images, estimate weight, and display prediction results. The webcam captures real-time images of coffee jars and is mounted on a tripod to ensure stable position-ing and optimal image acquisition. The estimated weight is displayed on the 1.3-inch OLED, directly connected to the Raspberry Pi, allowing quick access to weight information without additional devices. The hardware system design is illustrated in Fig. 9.

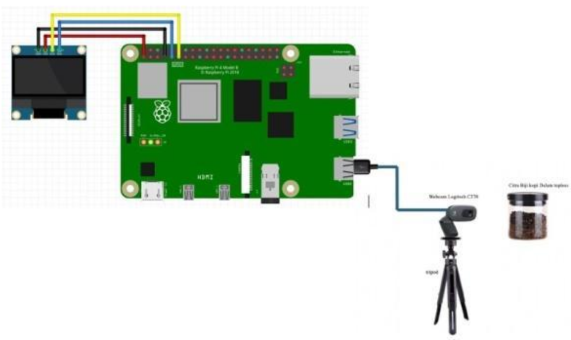


Fig. 9 Hardware design of the system

3.5 Software Design

The software design of the system consists of several main modules. First, image acquisition of coffee jars is performed using a webcam connected to the Raspberry Pi, capturing images periodically at the end of each operational day to serve as visual input data. Second, weight prediction is carried out using the MobileNet model, which has been trained on a dataset of coffee bean images with actual weight labels; the inference output provides an estimated weight in grams. Third, the predicted data is transmitted and stored in a database for automatic daily stock recording, facilitating inventory monitoring. Finally, the estimated results are displayed via a simple web interface, enabling the staff of Roetin Coffee Shop to evaluate raw material usage, plan purchases, and minimize waste.

3.6 System Testing

System testing was conducted to evaluate the overall performance of the image based coffee bean weight estimation system. The first test involved the confidence score, which aimed to ensure that the detection model could recognize objects with a high level of confidence in each prediction, both for container bounding boxes and coffee beans. The higher the confidence score, the greater the model's trust in its detection results. Subsequently, the weight prediction test was carried out by taking three measurements for each container using the webcam and CNN based system, and then comparing the results with the actual weight to calculate the average and the percentage of error. The test results indicate that the system is capable of providing weight estimates with a satisfactory level of accuracy according to the prior calibration.

Additionally, an integration test was performed for the website status, database storage, and OLED display. This test ensured that all functions operated smoothly, from coffee bean weight prediction to data storage in the database, and real-time visualization on the OLED screen. Based on the testing results, all system components demonstrated good performance, as indicated by the ”success” status on the website, correctly stored data in the database, and weight results displayed directly on the OLED screen, supporting automatic monitoring of coffee bean inventory.

4. Result and Discussion

4.1 Hardware Implementation Results

The hardware implementation of the coffee bean weight estimation system consists of two main parts, namely device design and component integration. The device is designed in the form of a compact enclosure that functions as a protective casing for all electronic components. The box is built with dimensions of approximately 14.5 cm × 9.5 cm × 5 cm, adjusted to the size of the Raspberry Pi 4 Model B which serves as the main processing unit. A dedicated slot is provided on the top side of the box for the installation of a 1.3-inch OLED display, which is used to present real-time coffee weight estimation results. The physical appearance of the hardware casing is shown in Fig. 10.

Fig. 10 Hardware en-closure design

Fig. 11 Integrated hardware assembly with enclosure

The hardware integration process is carried out by placing and connecting all electronic components inside the enclosure based on the predefined layout. The Raspberry Pi 4 Model B functions as the main controller responsible for executing the CNN inference process, handling network communication with the FastAPI server, and sending the estimated weight output to the OLED display. The webcam is connected to the Raspberry Pi via a USB port to capture coffee bean container images, while the OLED communicates using an I2C interface through GPIO 17 (SCL) and GPIO 27 (SDA). The integrated hardware assembly is presented in Fig. 11.

4.2 Software Implementation Results

The software implementation of the coffee bean weight estimation system consists of three main programs: the coffee bean weight estimation program, the FastAPI server with database storage, and the website dashboard. The estimation program was developed using the Python programming language on a Raspberry Pi, supported by OpenCV, TensorFlow, and luma.oled modules. The system captures images through a camera and detects the coffee beans and container using a pre-trained MobileNet-SSD model. Weight estimation is obtained by calculating the filling height of the container based on the vertical coordinate difference of the bounding box, then converting it into centimeters using a calibration factor stored in config.json. The height value is then processed using a linear interpolation method according to the calibration data of each container. The estimated height and weight are displayed through the OLED screen and periodically sent to the server, enabling automatic stock monitoring without the need for physical weight sensors.

Journal of Robotics, Automation, and Electronics Engineering
Vol. 4, No. 1, 2026, pp. 237-249
ISSN 3025-3780 (online)
ISSN 3025-4590 (printed)

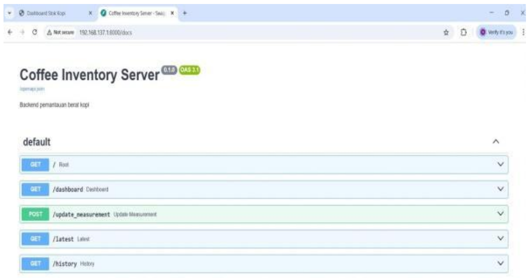


Fig. 12 Fast API server display

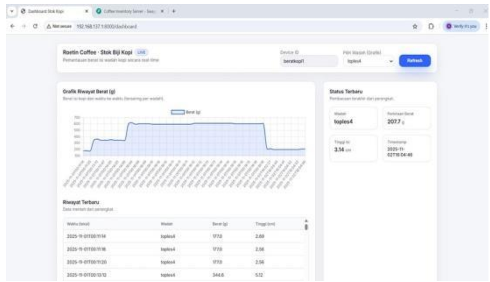


Fig. 13 Website dashboard for estimated coffee bean weight

The second program is the FastAPI server, which functions as a data communication handler between the Raspberry Pi and the website. The server receives estimated weight data via the HTTP protocol, stores it in an SQLite database, and distributes it to the web dashboard. The interface display of the server is shown in Fig. 12. The third program is the website display system which serves as a monitoring panel for coffee bean stock. The website presents information about container type, estimated height, estimated weight, as well as measurement history in table and graphic form to facilitate stock analysis. The website dashboard visualization is shown in Fig. 13.

4.3 Implementation of the CNN (Convolutional Neural Network) Method

The MobileNet SSD (Single Shot MultiBox Detector) method was implemented to detect container and coffee bean objects during the weight estimation process based on the height of the beans inside the container. MobileNet-SSD combines the lightweight MobileNetV2 architecture as the feature extractor with the SSD network as the object detector, enabling multi-object detection in a single inference.

This process began with training the MobileNet-SSD model using an image dataset obtained from six types of containers, consisting of four jar types (toples1, toples2, toples3, toples4) and two plastic pack types (plastik1kg and plastik2kg). Each image also contained a coffee bean object (bijikopi) used as an indicator of the container fill height. All images were annotated using Labeling in Pascal VOC (XML) format to define accurate bounding boxes and object classes.

The dataset was divided into 70% train set, 20% validation set, and 10% test set, with a total of 3,849 images: 2,694 training data, 769 validation data, and 386 test data. The dataset consisted of seven main classes: toples1, toples2, toples3, toples4, plastik1kg, plastik2kg, bijikopi. Model train-ing was conducted using the TensorFlow Object Detection API with the SSD MobileNetV2 FPNLite 320×320 architecture. The training configuration consisted of 25,000 steps and batch size = 4. The training process ran for approximately 21 hours on Google Colab CPU using Python 3.10 and TensorFlow 2.12.0.

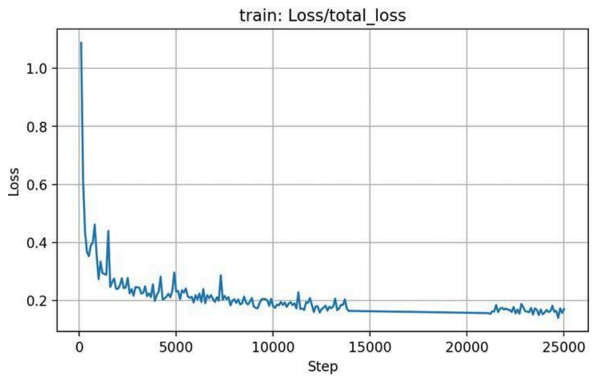


Fig. 14 Model training total loss graph

Based on the training graph, the initial loss value was very high because the weights were still random. As iterations increased, the loss decreased progressively until it stabilized below 0.2 around 20,000–25,000 steps, indicating that the model had successfully learned and reached convergence. The model evaluation results are shown through the mAP and classification performance metrics below:

- mAP@50:0.95 = **0.957**
- mAP@50 = **0.999**
- mAP@75 = **0.999**

• Recall = **0.971**

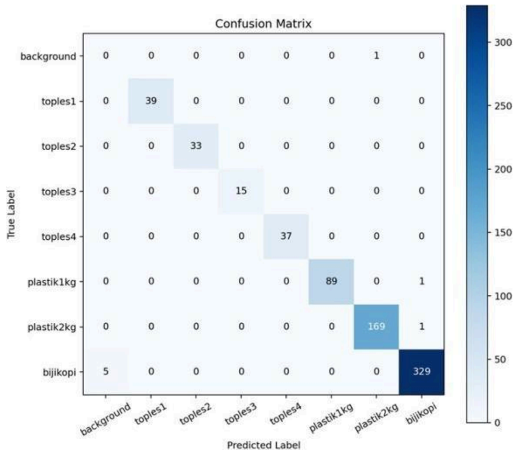


Fig. 15 Model confusion matrix results

The model was further evaluated using a confusion matrix for two major classification groups: coffee beans (bijikopi) and containers (jar/plastic). Precision, Recall, F1-Score, and Accuracy for each class are shown in Table 1. The results indicate that the model achieved very high performance with an overall accuracy of 0.994, showing that object detection is highly accurate and suitable for application in the weight estimation system.

Table 1. Precision, Recall, F1-Score, and Accuracy for Each Class

Class	Precision	Recall	F1-Score	Accuracy
toples1	1.000	1.000	1.000	1.000
toples2	1.000	1.000	1.000	1.000
toples3	1.000	1.000	1.000	1.000
toples4	1.000	1.000	1.000	1.000
plastik1kg	0.988	0.989	0.988	0.994
plastik2kg	0.994	0.994	0.985	0.985
bijikopi	0.994	0.994	0.985	0.985

(a) Original Image of Plastic Bag 1 kg

(b) Original Image of Plastic Bag 2 kg

(c) Webcam Detection Result of Plastic Bag 1 kg

(d) Webcam Detection Result of Plastic Bag 2 kg

Fig. 16 Capture and detection results for 1kg and 2kg plastic containers

During image capture, the camera position was adjusted to clearly display the plastic container outline and coffee surface. However, detection issues occurred with transparent jars, as the container boundary was difficult to read, resulting in inaccurate bounding boxes. The applied solution was adding a sticker behind

ISSN 3025-3780 (online)
ISSN 3025-4590 (printed)

the jar to increase edge contrast, allowing bounding box detection to become more stable and accurate.

4.4 Project Testing

Project testing was carried out to evaluate the system's ability in estimating coffee bean weight based on image processing, starting from object detection, weight prediction, accuracy calculation, to data integration with server and OLED display. The testing is divided into four main stages as follows.

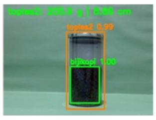
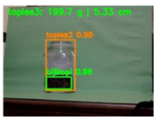
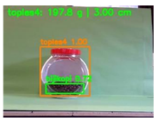
4.5.1 Confidence Score Testing

This test aims to determine the CNN model's capability in recognizing container and coffee bean objects through the confidence value on the bounding box. The higher the confidence value, the stronger the model certainty in detecting an object. The confidence score test results are shown in Table 2. Confidence values range from 0.68–1.00, indicating that object detection runs well across all test samples.

Fig. 17 Inaccurate bounding box on trans-parent jar

Fig. 18 Accurate bounding box after sticker placement

Table 2. Confidence Score Test Results on Bounding Box

Container	Image	Container Conf.	Coffee Bean Conf.
Jar 1		0.95	1.00
Jar 2		0.99	1.00
Jar 3		0.98	0.96
Jar 4		1.00	0.72
Plastic 1 Kg		0.96	1.00
Plastic 2 Kg		1.00	0.68

4.5.2 Coffee Bean Weight Prediction Testing

The testing was performed by comparing three measurement estimations (P1, P2, P3) for each container with the actual weight. The error percentage was calculated using Eq. 2.

$$\text{Error}(\%) = \frac{\text{Estimated Weight} - \text{Actual Weight}}{\text{Actual Weight}} \times 100\% \quad (2)$$

The testing summary is presented in Table 3. From a total of 18 samples, the system achieved an average error of only 0.91%.

Table 3. Coffee Bean Weight Prediction Testing

Container	Actual	P1	P2	P3	Avg	Error %
Jar 1	200	203.1	197.9	197.9	199.6	0.20
	400	400.9	400.9	400.9	400.9	0.22
	500	499.9	494.7	499.9	498.1	0.38
Jar 2	100	99.1	101.6	99.1	99.9	0.07
	200	205.9	208.9	205.9	206.9	3.45
	300	307.4	310.1	304.8	307.4	2.46
Jar 3	120	119.2	123.5	119.2	120.6	0.50
	300	312.5	304.5	304.5	307.1	2.36
	500	510.2	518.3	500.8	509.7	1.94
Jar 4	200	197.8	207.7	197.8	201.1	0.55
	350	344.6	353.5	353.5	350.5	0.14
	600	602.3	594.6	610.0	602.3	0.38
Plastic 1kg	100	100.2	97.9	100.2	99.4	0.60
	200	201.4	198.6	201.4	200.4	0.20
	300	312.4	303.0	298.4	304.6	1.53
Plastic 2kg	100	97.7	102.7	97.7	99.3	0.70
	500	507.9	498.2	498.2	501.4	0.28
	700	713.1	693.5	703.3	703.3	0.47
Average Error						**0.91**

4.5.3 Overall Accuracy Testing

System accuracy was calculated using Eq. 3.

$$\text{Accuracy}(\%) = \left(1 - \frac{|\text{Estimated Weight} - \text{Actual Weight}|}{\text{Actual Weight}}\right) \times 100\% \quad (3)$$

Table 4 shows that the system achieved an average accuracy of 99.09%.

Table 4. Overall Accuracy Testing

Container	Actual	Estimated	Accuracy (%)
Jar 1	200	199.6	99.60
	400	400.9	99.77
	500	499.9	99.98
Jar 2	100	99.9	99.93
	200	206.9	96.55
	300	307.4	97.53
Jar 3	120	120.6	99.50
	300	307.1	97.63
	500	509.7	98.06
Jar 4	200	201.1	99.45
	350	350.5	99.85
	600	602.3	99.61
Plastic 1kg	100	99.4	99.40
	200	200.4	99.80
	300	304.6	98.46
Plastic 2kg	100	99.3	99.30
	500	501.4	99.72
	700	703.3	99.52
Overall Accuracy			**99.09%**

5. Conclusion

Based on the process of designing, implementing, and testing a non-contact coffee bean weight measurement system using a webcam and a Convolutional Neural Network (CNN) with the MobileNet SSD architecture, the developed system successfully detected coffee beans and the container automatically and performed quantity estimation with an average accuracy of 99.09% and a maximum error of 3.45%. These results

ISSN 3025-3780 (online)
ISSN 3025-4590 (printed)

indicate that the system is capable of providing weight estimations that are very close to manual measurements, making it suitable as a supporting solution for stock opname activities at Roetin Coffee Shop. For further development, it is recommended to add an automatic calibration algorithm, increase dataset variations (lighting conditions, viewing angles, and container types), and integrate cloud storage services so that the system can operate more adaptively, accurately, and be accessed in real-time from various devices.

Author Contribution:

All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

Funding:

This research received no external funding.

Acknowledgment:

The author would like to express gratitude to Universitas Negeri Yogyakarta for providing research facilities and support.

Conflicts of Interest:

The authors declare no conflict of interest.

References

[1] Badan Pusat Statistik, Statistik Kopi Indonesia 2023. Jakarta, Indonesia: Badan Pusat Statistik, 2024, Publikasi No. 05100.24022, ISSN 2714-8505.

[2] K. Anam et al., Budidaya Tanaman Kopi dan Olahannya untuk Kesehatan. Makassar, Indonesia: CV Tohar Media, 2023, ISBN: 978-623-8148-42-4.

[3] J. G. Djoni and Tony, "Sistem manajemen stok barang berbasis web pada Toko Bangunan Sumber Abadi," Jurnal Ilmu Komputer dan Sistem Informasi, vol. 12, no. 1, 2024, doi: 10.24912/jiksi.v12i1.28273.

[4] Wahyudi, A. Rahman, and M. Nawawi, "Perbandingan nilai ukur sensor load cell pada alat penyortir buah otomatis terhadap timbangan manual," ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika, vol. 5, no. 2, pp. 207–220, 2017, doi: 10.26760/elkomika.v5i2.207.

[5] M. M. Purba and C. Rahmat, "Perancangan sistem informasi stok barang berbasis web di PT Mahesa Cipta," JSI (Jurnal Sistem Informasi) Universitas Suryadarma, vol. 8, no. 2, pp. 123–158, 2021, doi: 10.35968/jsi.v8i2.721.

[6] O. Irnawati, "Implementasi metode Waterfall pada sistem informasi stock opname," Indonesian Journal on Software Engineering (IJSE), vol. 4, no. 1, pp. 79–84, 2018, doi: 10.31294/ijse.v4i1.6301.

[7] S. B. Nauli and Riduan, "Perancangan aplikasi persediaan barang menggunakan barcode Quick Response dengan metode First-In First-Out berbasis mobile (Studi kasus: PT Kopi Kenangan)," SENTRI: Jurnal Riset Ilmiah, vol. 3, no. 4, pp. 2126–2137, 2024, doi: 10.55681/sentri.v3i4.2608.

[8] A. Fuadi and A. Suharso, "Perbandingan arsitektur MobileNet dan NASNetMobile untuk klasifikasi penyakit pada citra daun kentang," JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika), vol. 7, no. 3, pp. 701–710, 2022, doi: 10.29100/jipi.v7i3.3026.

[9] Herdianto and D. Nasution, "Implementasi metode CNN untuk klasifikasi objek," METHOMIKA: Jurnal Manajemen Informatika & Komputerisasi Akuntansi, vol. 7, no. 1, pp. 54–60, 2023, doi: 10.46880/jmika.Vol7No1.pp54-60.

[10] S. Ramdani and A. Rahmatulloh, "Implementasi MobileNet untuk klasifikasi gambar dan deteksi emosi menggunakan KERAS," JUSTIN (Jurnal Sistem dan Teknologi Informasi), vol. 12, no. 2, pp. 259–264, 2024, doi: 10.26418/justin.v12i2.73389.

[11] A. Wijaya, B. R. Daryono, R. Toyib, and Y. Apridiansyah, "Analisis pengenalan pola pada citra digital untuk prediksi berat buah sawit," Decode: Jurnal Pendidikan Teknologi Informasi, vol. 4, no. 3, pp. 713–724, 2024, doi: 10.51454/decode.v4i3.481.

[12] D. B. Lasfeto and M. D. Letik, “Rancangan teknologi pengukur berat badan ternak sapi Timor berbasis citra sebagai pengganti timbangan mekanis dalam mendukung inovasi peternakan sapi di Pulau Timor Provinsi Nusa Tenggara Timur,” in Proc. Seminar Nasional dan Konferensi Sistem Informasi, Informatika dan Komunikasi (SEMMAU 2015), Kupang, Indonesia, 2015, pp. 129–134.

[13] Martinus, A. Susilo, M. Telaumbanua, and M. A. Muhammad, “Rancang bangun sistem penghitung jumlah dan massa biji kopi berbasis mikrokontroler pada konveyor sabuk,” Barometer, vol. 5, no. 2, pp. 267–271, 2020, doi: 10.35261/barometer.v5i2.3816.

[14] I. A. Hidayat, “Implementasi algoritma FIFO pada sistem manajemen stok dan deteksi kadaluwarsa otomatis di Toko Agus Mart,” S1 thesis, Universitas Muhammadiyah Magelang, Magelang, Indonesia, 2025.