

## YOLO ALGORITHM-BASED VISITOR DETECTION SYSTEM FOR SMALL RETAIL STORES USING SINGLE BOARD COMPUTER

Tati Erlina<sup>\*1</sup>, Muhammad Fikri<sup>2</sup>

Department of Computer Engineering, Universitas Andalas, Indonesia<sup>12</sup>

tatierlina@it.unand.ac.id

Received : 25 March 2023, Revised: 19 May 2023, Accepted : 20 May 2023

\*Corresponding Author

---

### ABSTRACT

*In Indonesia, assistance for small enterprises has grown in recent years. However, a monitoring system is required to support these enterprises and ensure their expansion and survival. Using a single-board computer and the YOLO algorithm, we construct a visitor tracking system in this study to meet this demand. To capture objects and categorize them as human or non-human, we employ the YOLOv4-tiny model, which has a mAP of 89.21%. Human visitors are welcomed with the use of a speaker. A telegraph bot that notifies the owner of the retail establishment of the visitor's presence also makes the presumption as to whether the visitor is a potential customer or an intruder. Our research demonstrates that the created monitoring system effectively recognizes and categorizes visits, enabling retail store owners to make defensible choices regarding visitor interaction and security precautions. Small business owners can save personnel costs while still maintaining high levels of client engagement and security. The theoretical application of this research is the creation of a visitor monitoring system that is affordable and may be used in small enterprises, particularly in Indonesia. The practical ramifications of our research include the possibility for small retail business owners to boost profits by lowering labor expenses while raising customer satisfaction and security. The importance of our study lies in its role in creating a monitoring system that will support small enterprises and increase their sustainability.*

**Keywords:** Monitoring System, Small Retail Store, Raspberry Pi, YOLO

### 1. Introduction

Serving both urban and rural clientele, small retail establishments are a crucial part of Indonesia's cultural legacy and economic landscape. Despite the rise of e-commerce, local small-scale retail businesses are still thriving in many areas because of their attentive customer care and focus on the neighborhood. However, small business owners in Indonesia confront various difficulties, such as limited access to finance, labor, and technology (Hermawan & Nugraha, 2022; Maksum et al., 2020; Raharja et al., 2019). Since the economy changes so quickly, many small firms struggle to stay profitable and competitive.

The necessity to offer adequate security while limiting personnel expenses presents a significant barrier for small retail store operators. Store owners must balance the requirement for consumer involvement and accessibility with the necessity to avoid illegal access and loss. Theft and other security issues constantly worry store owners (Korgaonkar et al., 2021). The issue of retail business monitoring systems has been addressed in previous studies in several ways. Numerous studies have been conducted on the sector to increase effectiveness and consumer satisfaction. A 3D Vision-Based Shelf Monitoring system (3D-VSM) was suggested in one of these studies (Milella et al., 2021) in order to estimate the On-Shelf Availability (OSA) of goods in a retail setting. This system compares a reference model of the shelf with its actual status to offer up-to-date information about product availability for client purchase and create notifications on Out-Of-Stock (OOS) events. Based on biometric information and facial expressions, a different study (Generosi et al., 2018) developed an emotional tracking system to evaluate the shopping experience at several touchpoints in a retail store. The effectiveness of the system in detecting emotions and preventing sex, age, and ethnicity discrimination against customers was tested preliminarily by the study. An innovative retail monitoring system was proposed in a study (Jafriz & Mansor, 2022) based on the Intel Distribution of Open VINO toolkit. This system uses trained models and deep learning techniques to count people, people entering and leaving the premises automatically, and the distance between people to ensure social distance. Five trials were used to evaluate the system, which showed great accuracy and efficiency in counting and

recognizing persons and evaluating social distance. These solutions are less suited for small firms with limited resources because they can be pricey and challenging to install and maintain.

Using single-board computers and basic sensors to provide affordable and effective security monitoring is an alternate strategy that has lately gaining prominence. These devices can be programmed to detect motion (Babu et al., 2020; Guha et al., 2020; Mathur et al., 2017), sound (Bhambani et al., 2020; J. Kim et al., 2020). They can also detect temperature changes (Arun et al., 2020; Jadon et al., 2019; Jaihar et al., 2020; Priyanka et al., 2022). They can be linked to cameras to deliver video footage of the area in issue, enabling owners to watch over their shops remotely. Another benefit of using this kind of system is that the store owner may easily install and maintain it, eliminating the need for exorbitant installation or continuing maintenance fees. A complete security system suited to the requirements of small enterprises can be provided by integrating these devices with additional software and hardware options. Despite the potential advantages of this kind of security system, more investigation is required to comprehend its capabilities and limitations entirely and to create more user-friendly and effective solutions for small retail store owners.

We suggest an affordable visitor monitoring system that uses a single-board computer and the YOLO (You Only Look Once) algorithm to overcome this problem. The system detects and tracks things inside the predetermined border areas of the store using a webcam, Raspberry Pi, speaker, and push button. The system can distinguish between legitimate visitors and possible robbers thanks to the YOLO algorithm, which lowers false alarms and boosts overall security. The technology alerts the store owner to the visitor's presence and, if necessary, gives more footage and details. The primary goal of this project is to create a monitoring system that can aid small business owners in enhancing their security and lowering labor costs. Store owners can concentrate on offering top-notch customer service and running their businesses more effectively by offering an automated and trustworthy security system. The system is a practical option for small enterprises in Indonesia and other comparable environments due to its affordability and simplicity of usage.

Creating a low-cost visitor monitoring system that can improve the security of small retail establishments is one of the study's theoretical contributions. The possibility for small retail business owners to increase their bottom line by lowering labor expenses while boosting consumer engagement and security is one example of a practical contribution. The importance of our study lies in its role in creating a monitoring system that will support small enterprises and increase their sustainability.

## 2. Literature Review

In this part, we will perform a literature review of academic sources pertinent to our research issue. We want to comprehend better important ideas, advances, and findings connected to our study problem or subject. This review will give a thorough overview of the theories and studies that have already been done on our subject and identify any knowledge gaps that need to be filled. Convolutional neural networks (CNNs), You Only Look Once (YOLO), OpenCV, and single-board computers will be highlighted in this overview of machine learning. We will then concentrate on current studies about monitoring systems in retail establishments.

Computer vision is only one of the many industries transformed in recent years by cutting-edge technology. Machine learning is one of the most critical developments in this area, which entails teaching algorithms to recognize patterns and reach data-driven conclusions. Machine learning has shown to be particularly useful in image processing applications, such as estimating concrete surface roughness (Jiang et al., 2021; Protopapadakis et al., 2019; Valikhani et al., 2021), defect in additive manufacturing (Caggiano et al., 2019; Scime & Beuth, 2019; Wang et al., 2020) and bioimage analysis (Berg et al., 2019; Ma et al., 2021; Moen et al., 2019) where it can automate tasks that would be time-consuming or difficult for humans to perform.

A typical type of neural network used in machine learning for image and video processing applications is the convolutional neural network (CNN). Deep learning models, a machine learning method created to learn and extract high-level characteristics from data automatically, include CNNs as a subset. CNNs are particularly helpful for tasks like image recognition (Sim et al., 2019), object detection (Hashemzahi et al., 2020), and image segmentation (Sharma et al.,

2020) in the context of machine learning. The networks can automatically learn and extract edges, corners, and other essential features from images. These features can subsequently be applied to predictions or the classification of images. CNN uses convolutional layers to extract features from images, and fully linked layers are then used to categorize the objects in the image. A representative layer of the CNN architecture is shown in Fig. 1, demonstrating that the architecture comprises several convolutional layers, followed by several fully linked layers.

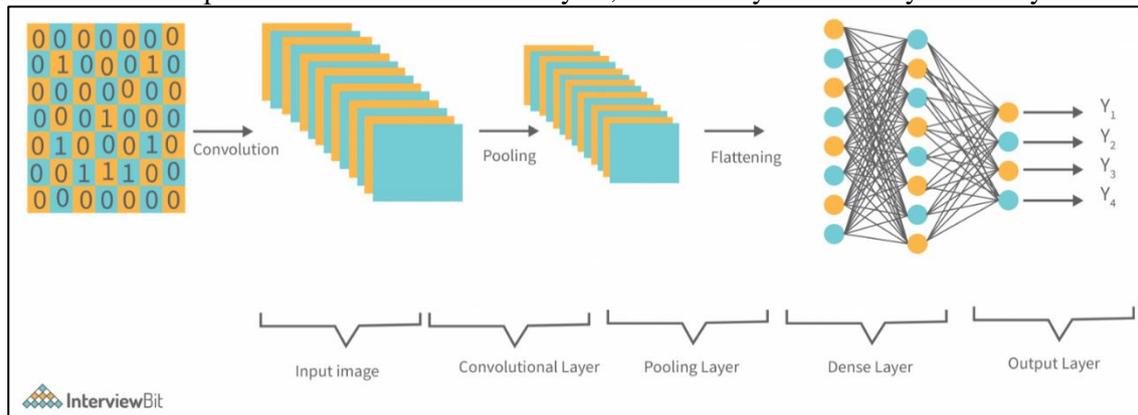


Fig. 1. Typical CNN Architecture

The well-known object recognition system YOLO uses convolutional neural networks (CNNs) to identify items in still and moving pictures. YOLO has emerged as one of the most popular object recognition algorithms in computer vision applications thanks to its real-time object detection capabilities (Ullah, 2020). The primary principle of this technique is to partition an input image into a grid of cells and run a CNN on each cell to identify objects (Redmon et al., 2020) This strategy is distinct from other object detection algorithms that look for things using sliding windows and region recommendations. The technique for detecting YOLO is shown in Figure 2. YOLO can detect objects in real-time without requiring computationally intensive procedures because CNNs are used to process each cell. YOLO also has the advantage of being able to detect many objects in a single pass, which is something that other algorithms cannot do.

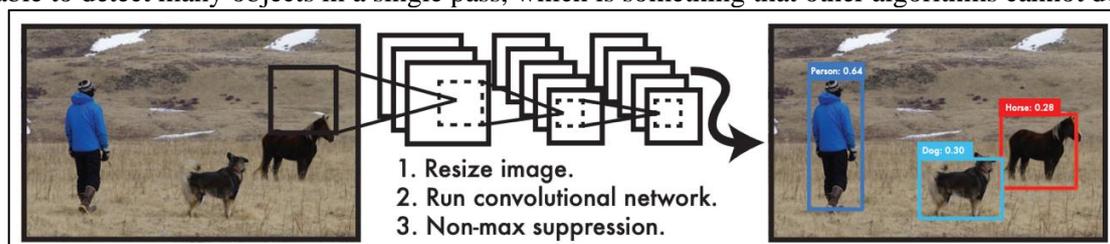
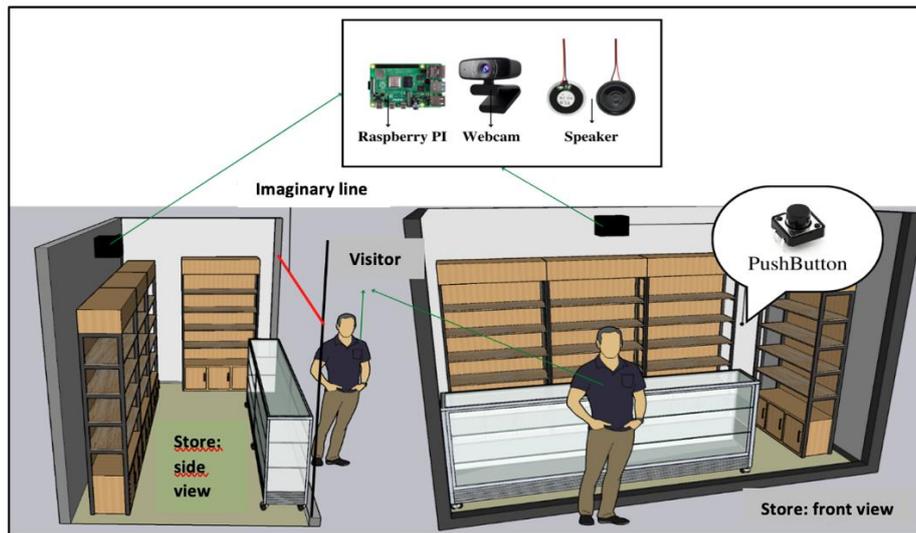


Fig. 2. The YOLO Detection System (Redmon et al., 2020)

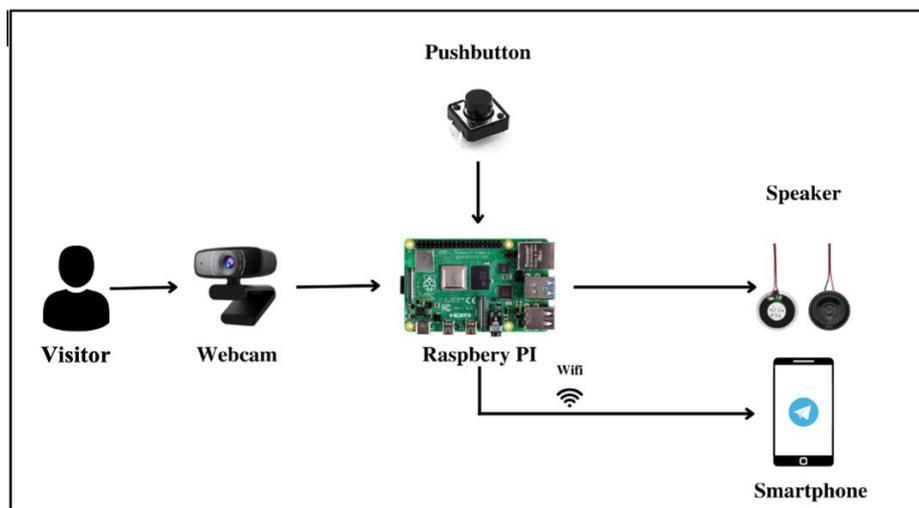
Open CV (Open Source Computer Vision), a collection of programming functions used to carry out various computer vision tasks, such as image and video analysis, object recognition, and tracking, is another crucial image processing tool employed in this work. Due to its adaptability, effectiveness, and simplicity, OpenCV is extensively used and compatible with several computer languages, including Python, C++, and Java (Gollapudi & Gollapudi, 2019). When used together, YOLO and OpenCV can build solid object detection systems that precisely identify things in real-time. These systems can automate operations, increase accuracy, and reduce time by utilizing machine learning and computer vision.

### 3. Research Methods

The research methodology section of this article outlines the approach taken to conduct the study and the methods used to collect and analyze data. The research starts with problem identification by exploring issues encountered by retail shop owners regarding their limitation to attend the shop full time and the need for the capability to leave the store to do personal matters without losing their potential customers, as well as keeping the store secure from theft. Based on the problems, a literature review is conducted to gather references such as journals or other related resources.



(a)



(b)

Fig. 3. General Design: (a) Context Of Usage (b) Hardware Scheme.

The next step is conducting a system requirement analysis to identify the system's functioning needs. Therefore, the system requirements must be determined based on the functional and non-functional needs of the system. The applicable conditions are the needs for the system to function correctly, in the case of this research, including the four following points: a) The system must be able to capture the appropriate object by positioning the tool so that other entities do not obstruct it. b) The system must have an excellent human object recognition model. c) The system must be connected to the internet to send data to Telegram. d) The owner's handheld must have the Telegram application installed to receive notifications from the system. The non-functional requirements, accordingly, are the needs that are not involved in the process of the system running, such as the system must be connected to electricity, and the system requires real-time processing.

To fulfill the requirement mentioned earlier, a set of hardware which consists of a Raspberry Pi, camera, push button, speaker, and android smartphone, are utilized. Further, on the software side, we exploit the YOLO library, OpenCV, and Telegram application. Regarding data, we use the Human Detection Dataset (<https://www.kaggle.com/datasets/constantinwerner/human-detectiondataset>).

The Human Detection Dataset is a collection of images and corresponding annotations created to train and evaluate computer vision models that can detect humans in images. The dataset consists of two classes: images with and without human objects. There are 921 images in

the dataset, with 559 images containing one or more human objects and 362 images without humans. The data has been split into two subsets: a training set, 80% of the data with human objects, and the rest 20% of the class for the validation set. The training set consists of 447 images used to train the model. The validation set consists of 112 images, which tune the model's hyperparameters and monitor its performance during training. The test set consists of 100 images, which are used to evaluate the final performance of the trained model. To create the annotations for the dataset, each image was manually labeled with bounding boxes around each human present in the image. The bounding boxes were defined using the top-left and bottom-right coordinates of the box. Additionally, each bounding box was assigned a label indicating whether it contained a human.

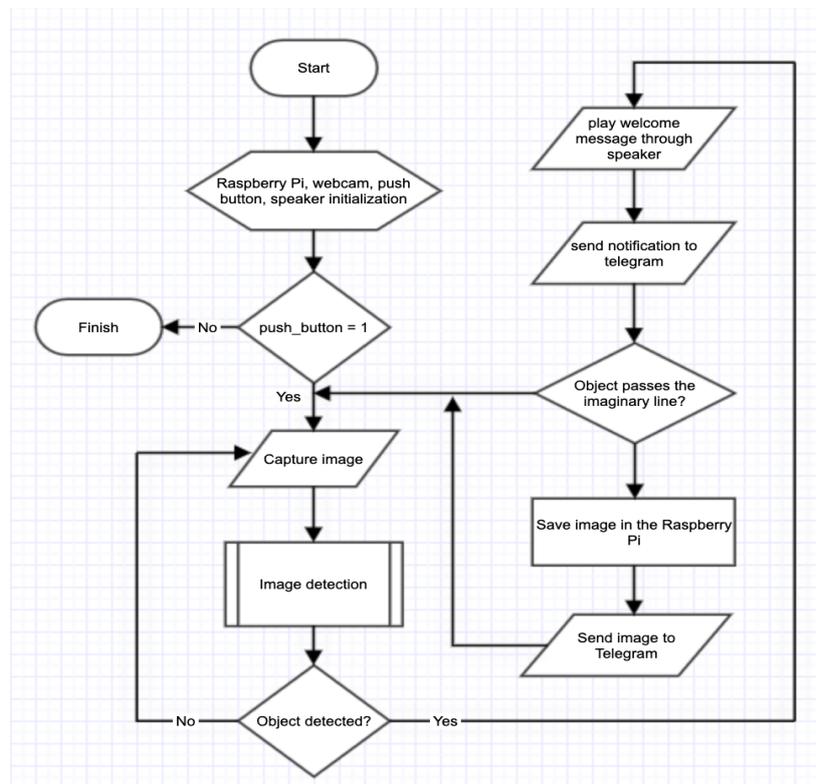


Fig. 4. Flowchart Process

After completing the system requirement analysis, a general design was created by integrating the components identified in the previous step. As illustrated in Fig. 3(a), the system is strategically placed to ensure precise object detection while avoiding obstacles. The webcam is positioned to detect the arrival of visitors. Suppose a visitor crosses the imaginary line defined in the software. In that case, the system identifies them as potential thieves and captures their image, which is then sent to the storekeeper's smartphone via Telegram. The imaginary line is set in the software to appear on the camera display, and it helps differentiate the zone that potential buyers cannot enter. The speaker and Raspberry Pi are located with the webcam to simplify system use. The push button, which functions as the device's power supply input, is also positioned for easy access by the store owner.

Figure 3 (b) shows the hardware scheme of the system. The system uses a webcam as its input device to capture images of human objects, which the Raspberry Pi then processes. Another input device is a pushbutton, which is the webcam's power supply. After the Raspberry Pi captures and processes the webcam input, the system sends the output to both a speaker and a telegram. The speaker provides information about when the shop owner will return, while the telegram output informs the storekeeper of other details. The system incorporates an imaginary line to distinguish between potential and non-buyers, sending two different output conditions to Telegram. If a potential buyer does not cross the imaginary line, the system sends only a notification to the storekeeper. On the contrary, if a potential buyer crosses the imaginary line,

the system captures an image and sends it directly to the storekeeper's Telegram. The overall flowchart process is shown in Fig. 4.

Hardware implementation of the design is shown in Fig. 5, in which the application results of the previously completed design are obtained. At the same time, the software implementation involves several steps. Firstly, the YOLO architecture is used for object detection in the system. Object detection is performed by training a model using YOLO, and the settings for the YOLO architecture are customized according to the system's needs. These settings include batch, subdivisions, width and height, max batch, steps, and filters.

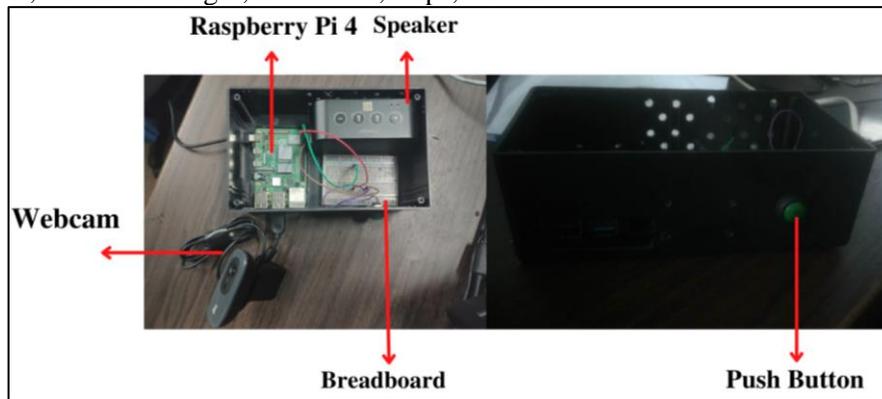


Fig. 5. Hardware Implementation

The YOLO model is obtained through training on Google Colab using Darknet. The images in the dataset are labeled, and the labeling results in a .txt file containing the coordinates of the part of the image to be detected. We measured a range of variables, as shown in Table 1, to capture the characteristics of the images and the human subjects they contain. The variables measured in the Human Detection Dataset were carefully selected to provide a diverse and representative set of images for training and evaluating computer vision models for human detection. The independent variables capture the variability in the images and the conditions under which they were captured. In contrast, the dependent variables provide accurate and reliable ground truth data for human detection. The independent variables include image size, camera resolution, lighting conditions, camera angle, and image background. These variables were selected to capture the variability in the images and ensure that the dataset contains a diverse range of images representative of real-world scenarios. The dependent variables we measured in the dataset include detection status, bounding box coordinates, number of humans detected, the pose of humans in the image, and the clothing of humans in the image. Detection status indicates whether a human is present in the image, while bounding box coordinates specify the location of the human(s) in the image. The number of humans detected provides information on the complexity of the images and the difficulty of the human detection task. The pose and clothing of the humans in the image provide additional information on the variability in the dataset and the challenges that must be addressed in human detection.

Table 1 - Variable Measured In Human Detection Dataset

Number	Independent Variable	Dependent Variable
1	Image size	Detection status
2	Camera resolution	Bounding box coordinates
3	Lightning condition	Number of human detected
4	Camera angle	Pose of human in image
5	Image background	Clothing of human in image

After obtaining the labeled dataset, the YOLO configuration is adjusted to the class for use. For this system, only one type, person, is used. The YOLOv4-tiny model is trained using the darknet framework with 2000 iterations for each class. During the training process, accuracy calculation was performed using the mean Average Precision (mAP) model. The mAP value is the result of calculating the accuracy of the trained object class, tested from validation data. The mAP value obtained in this model training is 89.21%. The model accuracy evaluation process begins with the first 1000 iterations and every 1000 iterations. The model is saved in a weight format file. In addition to obtaining the mAP value during the training process, the precision,

recall, and F-1 values are also accepted. Detailed information of the training results is shown in Fig 6.

```

calculation mAP (mean average precision)...
Detection layer: 30 - type = 28
Detection layer: 37 - type = 28
52
detections count = 260, unique_truth_count = 66
class_id = 0, name = Person, ap = 89.21% (TP = 56, FP = 15)

for conf_thresh = 0.25, precision = 0.79, recall = 0.85, F1-score = 0.82
for conf_thresh = 0.25, TP = 56, FP = 15, FN = 10, average IoU = 62.35 %

IoU threshold = 50 %, used Area-Under-Curve for each unique Recall
mean average precision (mAP@0.50) = 0.892138, or 89.21 %
Total Detection Time: 0 Seconds

Set -points flag:
^-points 101^ for MS COCO
^-points 11^ for PascalVOC 2007 (uncomment ^difficult^ in voc.data)
^-points 0^ (AUC) for ImageNet, PascalVOC 2010-2012, your custom dataset
    
```

Fig. 6. The yolo4-tiny Training Result

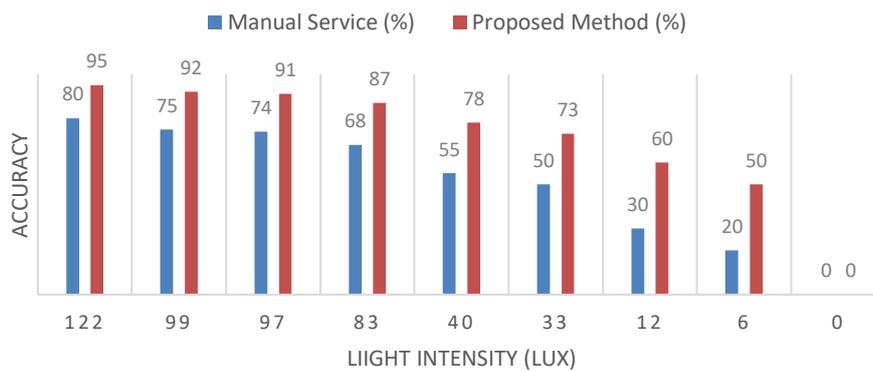


Fig. 7. The Comparison Between The Performance Of The Proposed Method And The Manual Service.

#### 4. Results and Discussions

Testing and analysis are performed to obtain results, determine the system's performance, and ensure that the system can function properly under certain conditions. The testing and commenting are divided into three parts: hardware testing, software testing, and system testing. Hardware testing is conducted to ensure that each hardware component can function correctly. Each element was tested individually by simulating possible scenarios that may occur when the system is running.

The camera is used as the input device for this system. The system detects each frame by matching the object responses in the camera with the pre-trained model. Objects are successfully detected, as evidenced by the appearance of a bounding box around the detected objects. This testing will focus on the system's ability to see things under different lighting conditions measured in lux units, ranging from the highest to the lowest or in low light conditions.

Table 2 - The Speed Of Data Transmission To Bot Telegram

Num.	Connection Source	Provider	Required time to send a text (s)	Required time to send an image (s)	Interval between customers (s)	Number of customers
1	Hotspot cellular	Telkomsel	1.21	5.17	10	5
2	Hotspot seluler	Telkomsel	1.30	5.63	12	4
3	Hotspot seluler	Telkomsel	1.41	4.60	15	4
4	Hotspot seluler	Telkomsel	1.24	14.4	8	6
5	Hotspot seluler	Telkomsel	1.34	4.75	11	4
6	WiFi	-	1.62	10.22	20	3
7	WiFi	-	1.41	5.81	13	4

Num.	Connection Source	Provider	Required time to send a text (s)	Required time to send an image (s)	Interval between customers (s)	Number of customers
8	WiFi	-	1.60	4.53	16	3
9	WiFi	-	1.55	13.22	9	6
10	WiFi	-	1.31	29.26	6	6



Fig. 8. The Telegram Bot Testing

Figure 7 shows the percentage of successful object detection for both the manual service and the proposed method at different light intensity levels, ranging from 122 Lux to 0 Lux. It shows that the proposed method generally outperforms the manual service regarding the percentage of successful object detection. At the highest light intensity level of 122 Lux, the manual service we have achieved an 80% success rate, while the proposed method achieved a 95% success rate. This trend continues as the light intensity decreases, with the proposed method consistently achieving more successful object detection than the manual service. It is worth noting that manual service may be subject to human error and can be influenced by factors such as fatigue, distractions, and personal biases.

The data transmission speed to Telegram was tested using several different network sources, namely cellular hotspots, WiFi, and LAN. Each of them was tested five times to obtain accurate results. The test results are presented in Table 2. The table shows the required time to send a text and an image using different connection sources and providers and the interval between customers and the number of customers. The table shows that the required time to send a text and an image varies depending on the connection source and provider. Generally, the WiFi connection appears slower than the cellular hotspot connection, with longer required times for sending a text and an image. Among the hotspot cellular connections, Trial 4 has a significantly longer required time to send an image than the others. The interval time between customers also varies, ranging from 6 to 20 seconds. The number of customers also varies, ranging from 3 to 7. These variables could potentially affect the model's performance in terms of speed, as more customers and shorter intervals between them could result in higher traffic and slower performance.

Figure 8 shows that the text and image messages have been successfully sent and displayed on Telegram. The message delivery can be delayed to avoid message stacking. The Telegram displays a notification of an incoming telegram message when a potential visitor arrives. When a potential visitor comes, it will be detected, and Raspberry Pi will send a message to Telegram. Suppose the potential visitor attempts to steal or crosses the predetermined imaginary line in the system. In that case, Raspberry Pi will capture an image, save it, and send it to the Telegram Bot.

The experiment results are presented in Table 3. The table shows the results of 6 experiments conducted under different conditions. The first three experiments were conducted to test the system's ability to detect visitors or potential buyers. The objects were set not to cross the imaginary line or were within an area visitors could occupy since the objects were not moving during the test. The following three experiments were conducted to test the system's ability to see thieves or objects crossing the imaginary line. The duration between messages and audio can be adjusted to avoid overlapping. The table shows that objects that cross the imaginary line

immediately saved in the Raspberry Pi's last frame and then sent to the Telegram Bot. In this condition, the system will not play audio through the speaker. Therefore, the system can differentiate between a thief and a visitor or potential buyer based on the object's movement toward the imaginary line that serves as the boundary.

The results of this study demonstrate the potential of a monitoring system to help small retail store owners in Indonesia address the challenge of providing adequate security while minimizing labor costs. By utilizing single-board computers, cameras, speakers, and other supporting hardware and software, the proposed system can automatically differentiate between prospective customers and potential thieves based on the store's defined border areas. In either case, the visitor would be automatically welcomed through a speaker and informed that the owner had been notified of their attendance. At the same time, the owner can receive notifications on whether the visitor crossed the predefined borderline, and other relevant footage can also be provided.

Table 3 - The Speed Of Data Transmission To Bot Telegram

Trial number	Object type	Speaker state	Detected image	Received information in Telegram
1	A person	Producing a voice		
2	A person	Producing a voice		
3	A person	Producing a voice		
4	A person	Producing a voice		
5	Multiple people	Producing a voice		
6	Multiple people	Producing a voice		

Our study aligns with prior research exploring various security and monitoring systems approaches. For example, the research in (Lohani et al., 2021; Vijverberg et al., 2014; Zhang et al., 2015) aims to identify unauthorized objects within a protected outdoor area during specific periods. The unique challenges due to the use of the outdoor environment in this research, such as then eliminated by indoor changing weather conditions, fluctuating light levels, and the presence of insects and animals, are addressed by the indoor system (Matern et al., 2013; Villamizar et al., 2018). Another previous research use video anomaly detection, which identifies unusual attributes in appearance or motion within recorded videos (Feng et al., 2021; Li et al., 2022). Some other studies have even created datasets containing anomalous activities and utilized

multiple learning instances for anomaly detection (Sultani et al., 2018). Additionally, the system (Cermeño et al., 2018; Nayak et al., 2021) explicitly focuses on detecting intrusions caused by human activities such as walking or driving a car. In this case, video acquisition must be conducted at 5 to 25 frames per second. In (Aravamuthan et al., 2020; S.-Y. Kim et al., 2013; Shao et al., 2014), improved intrusion detection is achieved by using additional sensors to provide depth information.

While these studies offer valuable insights into leveraging technology to address security challenges for small retail store owners, they may not be feasible for businesses with limited resources due to their high cost and complexity. In contrast, our proposed monitoring system is more affordable, easy to install, and maintain, making it a practical solution for small retail store owners in Indonesia. Furthermore, the system has the potential to enhance customer engagement by offering automated greetings to prospective customers and gathering relevant data on foot traffic and customer behavior.

It is important to note that while the proposed monitoring system has the potential to address the security challenges faced by small retail store owners in Indonesia, further research is needed to evaluate the system's effectiveness and usability in real-world settings. In particular, future studies could explore the system's impact on customer satisfaction and reduce theft and other security threats. Additionally, it may be beneficial to investigate other ways to reduce the system's cost and complexity to make it more accessible to a broader range of small retail store owners.

## 5. Conclusion

In this study, we build a monitoring system for small retail stores with the YOLO algorithm using a single-board computer, Raspberry Pi. Based on the implementation and testing of the visitor monitoring system with the YOLO algorithm using a single board computer, the following conclusions can be drawn: Firstly, the system can detect human objects and classify them as humans. The captured object images from the webcam are processed on the Raspberry Pi, and the system can differentiate between human and non-human objects based on the trained model. Secondly, the Yolov4-tiny method in this system can detect objects according to the training, with an mAP model value of 89.21%. The system can detect objects according to the training target and has high accuracy. Thirdly, the system can distinguish between visitors, potential buyers, and thieves based on object movement detection. Objects not crossing the area boundary will be classified as visitors, while objects detected crossing the border or imaginary line will be called thieves. Finally, the Telegram application can receive and display the results of object detection processing performed on this system. In addition, the speaker successfully plays audio when it meets certain conditions.

## References

- Aravamuthan, G., Rajasekhar, P., Verma, R. K., Shrikhande, S. V., Kar, S., & Babu, S. (2020). Physical intrusion detection system using stereo video analytics. *Proceedings of 3rd International Conference on Computer Vision and Image Processing: CVIP 2018, Volume 2*, 173–182. [https://doi.org/10.1007/978-981-32-9291-8\\_15](https://doi.org/10.1007/978-981-32-9291-8_15)
- Arun, M., Baraneetharan, E., Kanchana, A., & Prabu, S. (2020). Detection and monitoring of the asymptotic COVID-19 patients using IoT devices and sensors. *International Journal of Pervasive Computing and Communications*, 18(4), 407–418. <https://doi.org/10.1108/IJPCC-08-2020-0107>
- Babu, S., Pragathi, B. S., Chinthala, U., & Maheshwaram, S. (2020). Subject Tracking with Camera Movement Using Single Board Computer. *2020 IEEE-HYDCON*, 1–6. <https://doi.org/10.1109/HYDCON48903.2020.9242811>
- Berg, S., Kutra, D., Kroeger, T., Straehle, C. N., Kausler, B. X., Haubold, C., Schiegg, M., Ales, J., Beier, T., & Rudy, M. (2019). Ilastik: interactive machine learning for (bio) image analysis. *Nature Methods*, 16(12), 1226–1232. <https://doi.org/10.1038/s41592-019-0582-9>
- Bhambani, K., Jain, T., & Sultanpure, K. A. (2020). Real-time face mask and social distancing violation detection system using yolo. *2020 IEEE Bangalore Humanitarian Technology Conference (B-HTC)*, 1–6. DOI: 10.1109/B-HTC50970.2020.9297902

- Caggiano, A., Zhang, J., Alfieri, V., Caiazzo, F., Gao, R., & Teti, R. (2019). Machine learning-based image processing for on-line defect recognition in additive manufacturing. *CIRP Annals*, 68(1), 451–454. <https://doi.org/10.1016/j.cirp.2019.03.021>
- Cermeño, E., Pérez, A., & Sigüenza, J. A. (2018). Intelligent video surveillance beyond robust background modeling. *Expert Systems with Applications*, 91, 138–149. <https://doi.org/10.1016/j.eswa.2017.08.052>
- Feng, J.-C., Hong, F.-T., & Zheng, W.-S. (2021). Mist: Multiple instance self-training framework for video anomaly detection. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14009–14018. DOI: 10.1109/CVPR46437.2021.01379
- Generosi, A., Ceccacci, S., & Mengoni, M. (2018). A deep learning-based system to track and analyze customer behavior in retail store. *2018 IEEE 8th International Conference on Consumer Electronics-Berlin (ICCE-Berlin)*, 1–6. <https://doi.org/10.1109/ICCE-Berlin.2018.8576169>
- Gollapudi, S., & Gollapudi, S. (2019). OpenCV with Python. *Learn Computer Vision Using OpenCV: With Deep Learning CNNs and RNNs*, 31–50. [https://doi.org/10.1007/978-1-4842-4261-2\\_3](https://doi.org/10.1007/978-1-4842-4261-2_3)
- Guha, S., Chakrabarti, A., Biswas, S., & Banerjee, S. (2020). Implementation of Face Recognition Algorithm on a Mobile Single Board Computer for IoT Applications. *2020 IEEE 17th India Council International Conference (INDICON)*, 1–5. <https://doi.org/10.1109/INDICON49873.2020.9342290>
- Hashemzahi, R., Mahdavi, S. J. S., Kheirabadi, M., & Kamel, S. R. (2020). Detection of brain tumors from MRI images base on deep learning using hybrid model CNN and NADE. *Biocybernetics and Biomedical Engineering*, 40(3), 1225–1232. <https://doi.org/10.1016/j.bbe.2020.06.001>
- Hermawan, M. S., & Nugraha, U. (2022). The development of Small-Medium Enterprises (SMEs) and the role of digital ecosystems during the COVID-19 pandemic: A case of Indonesia. In *Handbook of Research on Current Trends in Asian Economics, Business, and Administration* (pp. 123–147). IGI Global. DOI: 10.4018/978-1-7998-8486-6.ch007
- Jadon, A., Omama, M., Varshney, A., Ansari, M. S., & Sharma, R. (2019). FireNet: a specialized lightweight fire & smoke detection model for real-time IoT applications. *ArXiv Preprint ArXiv:1905.11922*. <https://doi.org/10.48550/arXiv.1905.11922>
- Jafriz, I. Z., & Mansor, S. (2022). Smart Retail Monitoring System using Intel OpenVINO Toolkit. *International Journal of Technology*, 13(6), 1241. <https://doi.org/10.14716/ijtech.v13i6.5872>
- Jaihar, J., Lingayat, N., Vijaybhai, P. S., Venkatesh, G., & Upla, K. P. (2020). Smart home automation using machine learning algorithms. *2020 International Conference for Emerging Technology (INCET)*, 1–4. DOI: <https://doi.org/10.1109/INCET49848.2020.9154007>
- Jiang, Y., Pang, D., & Li, C. (2021). A deep learning approach for fast detection and classification of concrete damage. *Automation in Construction*, 128, 103785. <https://doi.org/10.1016/j.autcon.2021.103785>
- Kim, J., Min, K., Jung, M., & Chi, S. (2020). Occupant behavior monitoring and emergency event detection in single-person households using deep learning-based sound recognition. *Building and Environment*, 181, 107092. <https://doi.org/10.1016/j.buildenv.2020.107092>
- Kim, S.-Y., Kim, M., & Ho, Y.-S. (2013). Depth image filter for mixed and noisy pixel removal in RGB-D camera systems. *IEEE Transactions on Consumer Electronics*, 59(3), 681–689. DOI: <https://doi.org/10.1109/TCE.2013.6626256>
- Korgaonkar, P., Becerra, E. P., Mangleburg, T., & Bilgihan, A. (2021). Retail employee theft: When retail security alone is not enough. *Psychology & Marketing*, 38(5), 721–734. <https://doi.org/10.1002/mar.21460>
- Li, S., Liu, F., & Jiao, L. (2022). Self-training multi-sequence learning with transformer for weakly supervised video anomaly detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(2), 1395–1403. <https://doi.org/10.1016/j.ssci.2015.01.013>
- Lohani, D., Crispim-Junior, C., Barthélemy, Q., Bertrand, S., Robinault, L., & Tougne, L. (2021). Spatio-temporal convolutional autoencoders for perimeter intrusion detection.

- Reproducible Research in Pattern Recognition: Third International Workshop, RRPR 2021, Virtual Event, January 11, 2021, Revised Selected Papers*, 47–65. [https://doi.org/10.1007/978-3-030-76423-4\\_4](https://doi.org/10.1007/978-3-030-76423-4_4)
- Ma, X., Niu, Y., Gu, L., Wang, Y., Zhao, Y., Bailey, J., & Lu, F. (2021). Understanding adversarial attacks on deep learning based medical image analysis systems. *Pattern Recognition*, *110*, 107332. <https://doi.org/10.1016/j.patcog.2020.107332>
- Maksum, I. R., Rahayu, A. Y. S., & Kusumawardhani, D. (2020). A social enterprise approach to empowering micro, small and medium enterprises (SMEs) in Indonesia. *Journal of Open Innovation: Technology, Market, and Complexity*, *6*(3), 50. <https://doi.org/10.3390/joitmc6030050>
- Matern, D., Condurache, A. P., & Mertins, A. (2013). Automated Intrusion Detection for Video Surveillance Using Conditional Random Fields. *MVA*, 298–301.
- Mathur, S., Subramanian, B., Jain, S., Choudhary, K., & Prabha, D. R. (2017). Human detector and counter using raspberry Pi microcontroller. *2017 Innovations in Power and Advanced Computing Technologies (i-PACT)*, 1–7. DOI: 10.1109/IPACT.2017.8244984
- Milella, A., Marani, R., Petitti, A., Cicirelli, G., & D’Orazio, T. (2021). 3d vision-based shelf monitoring system for intelligent retail. *Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10–15, 2021, Proceedings, Part II*, 447–459. [https://doi.org/10.1007/978-3-030-68790-8\\_35](https://doi.org/10.1007/978-3-030-68790-8_35)
- Moen, E., Bannon, D., Kudo, T., Graf, W., Covert, M., & Van Valen, D. (2019). Deep learning for cellular image analysis. *Nature Methods*, *16*(12), 1233–1246. <https://doi.org/10.1038/s41592-019-0403-1>
- Nayak, R., Pati, U. C., & Das, S. K. (2021). A comprehensive review on deep learning-based methods for video anomaly detection. *Image and Vision Computing*, *106*, 104078. <https://doi.org/10.1016/j.imavis.2020.104078>
- Priyanka, J. S., Kiran, M. S., & Nalla, P. (2022). A Secured IoT-Based Health Care Monitoring System Using Body Sensor Network. In *Emergent Converging Technologies and Biomedical Systems: Select Proceedings of ETBS 2021* (pp. 483–490). Springer. [https://doi.org/10.1007/978-981-16-8774-7\\_39](https://doi.org/10.1007/978-981-16-8774-7_39)
- Protopapadakis, E., Voulodimos, A., Doulamis, A., Doulamis, N., & Stathaki, T. (2019). Automatic crack detection for tunnel inspection using deep learning and heuristic image post-processing. *Applied Intelligence*, *49*, 2793–2806. <https://doi.org/10.1007/s10489-018-01396-y>
- Raharja, S. J., Kostini, N., Muhyi, H. A., & Rivani. (2019). Utilisation analysis and increasing strategy: e-commerce use of SMEs in Bandung, Indonesia. *International Journal of Trade and Global Markets*, *12*(3–4), 287–299. <https://doi.org/10.1504/IJTGM.2019.101557>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2020). You only look once: Unified, real-time object detection. arXiv 2015. *ArXiv Preprint ArXiv:1506.02640*. <https://doi.org/10.1109/CVPR.2016.91>
- Scime, L., & Beuth, J. (2019). Using machine learning to identify in-situ melt pool signatures indicative of flaw formation in a laser powder bed fusion additive manufacturing process. *Additive Manufacturing*, *25*, 151–165. <https://doi.org/10.1016/j.addma.2018.11.010>
- Shao, L., Han, J., Kohli, P., & Zhang, Z. (2014). *Computer vision and machine learning with RGB-D sensors* (Vol. 20). Springer. <https://doi.org/10.1007/978-3-319-08651-4>
- Sharma, P., Berwal, Y. P. S., & Ghai, W. (2020). Performance analysis of deep learning CNN models for disease detection in plants using image segmentation. *Information Processing in Agriculture*, *7*(4), 566–574. <https://doi.org/10.1016/j.inpa.2019.11.001>
- Sim, H. S., Kim, H. I., & Ahn, J. J. (2019). Is deep learning for image recognition applicable to stock market prediction? *Complexity*, *2019*. <https://doi.org/10.1155/2019/4324878>
- Sultani, W., Chen, C., & Shah, M. (2018). Real-world anomaly detection in surveillance videos. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6479–6488. DOI: <https://doi.org/10.1109/CVPR.2018.00678>
- Ullah, M. B. (2020). CPU based YOLO: A real time object detection algorithm. *2020 IEEE Region 10 Symposium (TENSYP)*, 552–555. DOI: 10.1109/TENSYP50017.2020.9230778

- Valikhani, A., Jaberi Jahromi, A., Pouyanfar, S., Mantawy, I. M., & Azizinamini, A. (2021). Machine learning and image processing approaches for estimating concrete surface roughness using basic cameras. *Computer-Aided Civil and Infrastructure Engineering*, 36(2), 213–226. <https://doi.org/10.1111/mice.12605>
- Vijverberg, J. A., Janssen, R. T. M., de Zwart, R., & de With, P. H. N. (2014). Perimeter-intrusion event classification for on-line detection using multiple instance learning solving temporal ambiguities. *2014 IEEE International Conference on Image Processing (ICIP)*, 2408–2412. DOI: <https://doi.org/10.1109/ICIP.2014.7025487>
- Villamizar, M., Martínez-González, A., Canévet, O., & Odobez, J.-M. (2018). Watchnet: Efficient and depth-based network for people detection in video surveillance systems. *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 1–6. DOI: <https://doi.org/10.1109/AVSS.2018.8639165>
- Wang, C., Tan, X. P., Tor, S. B., & Lim, C. S. (2020). Machine learning in additive manufacturing: State-of-the-art and perspectives. *Additive Manufacturing*, 36, 101538. <https://doi.org/10.1016/j.addma.2020.101538>
- Zhang, Y.-L., Zhang, Z.-Q., Xiao, G., Wang, R.-D., & He, X. (2015). Perimeter intrusion detection based on intelligent video analysis. *2015 15th International Conference on Control, Automation and Systems (ICCAS)*, 1199–1204. DOI: <https://doi.org/10.1109/ICCAS.2015.7364811>