Automatic Retail Checkout System Using YOLO-Based Product Detection and Face Recognition

1st Ananda Agung Ismail
dept. of Information Technology
Universitas Muhammadiyah Yogyakarta
Yogyakarta, Indonesia
ananda.agung.ft23@mail.umy.ac.id

2nd Slamet Riyadi*
dept. of Information Technology
Universitas Muhammadiyah Yogyakarta
Yogyakarta, Indonesia
riyadi@umy.ac.id

***Abstract*— The rapid advancement of smart retail technologies has intensified the demand for automated and cashier-less checkout systems that can reduce transaction time and enhance customer experience. Traditional checkout methods, such as barcode scanning and manual point-of-sale operations, are often inefficient and susceptible to human error. This study proposes a smart retail checkout system that integrates computer vision–based object detection and biometric face recognition to enable seamless, contactless transactions. A YOLO-based deep learning model is employed to automatically detect and classify multiple retail products in real time, while an InsightFace-based face recognition framework is used to authenticate customers and associate transactions with registered user identities for payment verification. Experimental results demonstrate that the proposed system is capable of accurately recognizing visually similar products and reliably identifying customers under real-world retail conditions. The integration of object detection and face recognition into a unified end-to-end system significantly reduces checkout time, minimizes human intervention, and improves overall operational efficiency, highlighting its potential for practical deployment in smart retail environments..**

Keywords*— *Computer Vision, Face Recognition, Retail Checkout System, Object Detection, Deep Learning, Smart Retail, Cashier-less Checkout

I. INTRODUCTION

The rapid growth of smart retail technologies has increased interest in automated and cashier-less checkout systems aimed at reducing transaction time and improving customer experience. Traditional checkout methods, including barcode scanning and manual point-of-sale interactions, are time-consuming and prone to human error, creating operational inefficiencies especially in high-traffic environments [1], [2]. These limitations motivate the development of more efficient and user-friendly checkout solutions.

Existing automated checkout strategies typically rely on technologies such as QR code scanning or RFID tags for product identification. QR-based systems depend on stable network connectivity and proper QR presentation by users, which can introduce scanning delays or errors [3]. RFID-based solutions can accelerate transactions but involve significant infrastructure costs and raise security concerns regarding unauthorized tag usage [4], [5]. Additionally, traditional loyalty and membership programs still require customers to present physical cards or input membership numbers, which can be forgotten or misplaced, limiting seamless personalized checkout experiences [7], [8].

To address product identification challenges, computer vision-based object detection has emerged as a key technology enabling automatic recognition of retail products without physical tags. Deep learning object detection models such as YOLO (You Only Look Once) have demonstrated strong performance in identifying multiple objects in real time, making them well-suited to retail checkout applications [1], [2], [6]. However, retail environments present specific challenges, including visually similar products, occlusions, and varying lighting conditions, which can reduce detection accuracy in unconstrained real-world settings [1], [2]. Recent studies show that fine-tuning YOLO models and applying data augmentation can significantly enhance detection accuracy up to ~95% on benchmark retail datasets [2], [4].

While object detection addresses automatic product recognition, user authentication at checkout remains a separate challenge. Biometric payment methods, including face recognition, offer contactless and device-free user identification, eliminating the need for physical membership cards or wallets [7], [8], [10]. Face recognition frameworks such as InsightFace enable efficient real-time face detection and authentication [10], [11], and have been applied in retail to enhance transaction speed, security, and customer convenience [7], [9].

Despite advances in both object detection for products and face recognition for biometric authentication, few studies integrate these two components into a unified end-to-end system for automated retail checkout. Most prior work has focused either on improving product detection accuracy [1]–[5] or evaluating face recognition performance in isolation [7]–[12], without simultaneously optimizing both tasks within a shared retail pipeline. This integration gap limits the practical deployment of fully automated cashier-less checkout systems that can identify products and authenticate users seamlessly.

Therefore, this research proposes a smart retail checkout system integrating YOLO-based object detection with InsightFace-based face recognition, enabling simultaneous product identification and biometric customer authentication. The system aims to reduce checkout time, eliminate manual scanning and membership card interactions, and improve both accuracy and user experience in smart retail environments.

II. METHODOLOGY

A. Research Framework

The method employed to recognize and classify the Retail product brand utilizes the YOLOv12m pre-trained model and

recognize buyer face with insightface model. This framework is designed to provide a comprehensive, end-to-end overview of the entire process, which is illustrated in Figure 1.

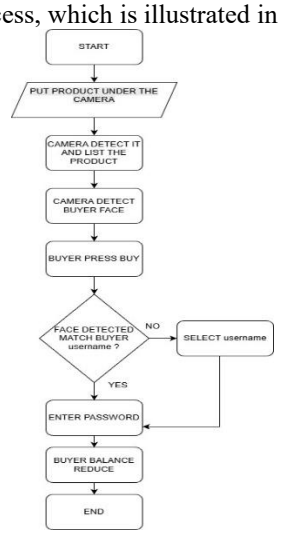

Figure 1 Flowchart Model

B. *Dataset and Pre-procesing*

The foundational component of the framework illustrated in Figure 1 is the workflow of the app. A crucial phase in this research was the systematic preparation of this data to ensure its quality and suitability for the YOLO models. Figure 2 illustrates the YOLO product training workflow for this phase, from dataset acquisition to testing the model readiness. The process was organized into four sequential stages: dataset selection and annotation, augmentation, training and data testing.

The dataset acquisition process was conducted using a webcam-based image capture system to collect real-world product images under practical operating conditions. Each product was placed under the camera and captured multiple times while being rotated and repositioned to obtain diverse visual perspectives.

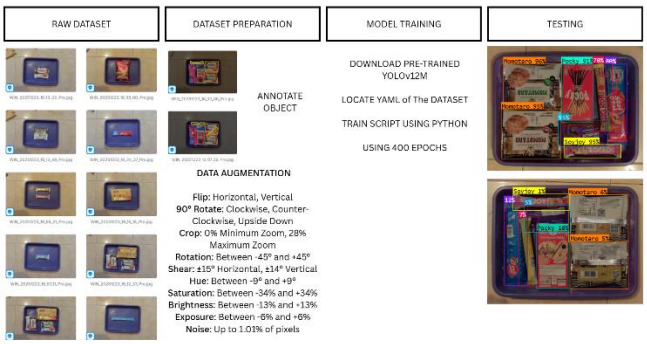

Fig 2. Flow Pre-processing Diagram

1) *Dataset Selection and Verification*

This study utilized the Indonesian product *"jajan"* dataset, a self collected photo. The dataset consists of 11 classes, each representing a brand of the *"jajan"* product, with sample images shown in Figure 3.

Fig 3. jajan Dataset

For the purpose of this research, the dataset was divided into a training subset consisting of 210 images (approximately 87.14%) and a validation subset consisting of 31 images (approximately 12.86%). The data were organized into directories containing an *images/* folder for image files and a *labels/* folder for YOLO-formatted bounding box annotations stored in *.txt* files. The overall dataset configuration, including dataset paths and the 11 class labels, was defined in a *data.yaml* file, enabling the YOLO model to correctly interpret the dataset structure and class definitions.

To enhance model robustness and generalization, data augmentation was applied during the training phase. Although the original dataset contained 210 training images and 31 validation images, the use of augmentation effectively increased the diversity of the training data. As a result, the training set yielded an estimated equivalent of approximately 800–1,200 unique training samples.

2) *Data Pre-processing and Augmentation*

Prior to training, all images in the training and validation sets were subjected to essential pre-processing steps. Each image was resized to a standardized resolution of 640×640 pixels to align with the input requirements of the YOLO architecture and to ensure uniformity during batch processing. Furthermore, image pixel values were normalized to a [0.0, 1.0] range, which is commonly applied in deep learning to stabilize optimization and improve training efficiency.

To enhance model robustness and minimize overfitting, data augmentation was applied dynamically to the training data. Considering the visual diversity of the "jajan" dataset—such as varying packaging orientations, illumination conditions, and camera perspectives—a comprehensive set of augmentation techniques was employed.

Data augmentation strategy :

- Flip Operations: Random horizontal and vertical flips were used to simulate different viewing angles.
- Rotation:
  Fixed rotations at 90° (clockwise, counter-clockwise, and inverted).
  Random rotations within a range of −45° to +45° to represent natural camera misalignment.
- Cropping and Zooming: Images were randomly cropped with zoom levels ranging from 0% to 28%, allowing the model to learn from partially visible objects.

- Shear Transformations: Horizontal shear up to ±15° and vertical shear up to ±14° were applied to emulate perspective distortion.
- Noise Injection: Random noise affecting up to 1.01% of image pixels was introduced to simulate sensor noise and image degradation.
- Color Adjustments:
  Hue variations between −9° and +9°.
  Saturation changes within −34% to +34%.
  Brightness adjustments between −13% and +13%.
  Exposure variations within −6% to +6%.

3) Model Training Parameters

The model was trained using the YOLOv12m architecture initialized with pretrained weights to accelerate convergence. Training was conducted on the prepared dataset using the ultralytics YOLO training pipeline. All experiment were executed with identical training settings to ensure consistency and reproducebility.

The model was trained for 600 epochs, allowing sufficient iterations for learning complex visual patterns present in the *jajan* dataset. A dynamic batch size configuration was employed (batch = 0.90), enabling automatic adjustment based on available GPU memory to maximize hardware utilization. Data loading was parallelized using 8 worker threads to improve training efficiency.

C. YOLO Modelling

YOLO (You Only Look Once) is a well-established real-time object detection framework that predicts bounding boxes and class probabilities in a single forward pass. YOLO has undergone continuous development since its inception, with each iteration introducing architectural and optimization improvements aimed at increasing detection accuracy while maintaining computational efficiency.

In this research, YOLO models are used to improve object detection on the available dataset. The choice of models is based on achieving a balance between inference speed and prediction accuracy.

YOLO12 represents a new generation of YOLO models characterized by an attention-centric architecture that extends beyond traditional convolutional designs while retaining real-time inference efficiency. At its core, YOLO12 integrates novel mechanisms such as Area Attention, which processes large receptive fields efficiently by dividing feature maps into regions, allowing richer contextual representation without incurring the full cost of standard self-attention. YOLO12 also incorporates Residual Efficient Layer Aggregation Networks (R-ELAN), which enhance feature aggregation and stabilize training for attention-based models. Additionally, optimized attention design—including use of FlashAttention and the removal of positional encodings—further improves computational efficiency and compatibility with the YOLO framework.

D. Performance Evaluation Model YOLO

The performance of the YOLOv12m model is evaluated using Accuracy, Precision, Recall, F1-Score, mean Average Precision (mAP), Intersection over Union (IoU), and Confusion Matrix analysis. Accuracy measures overall prediction correctness, Precision reflects the reliability of positive detections, and Recall indicates the model's ability to detect all relevant objects, with the F1-Score balancing Precision and Recall. mAP evaluates detection performance across multiple IoU thresholds, while the Confusion Matrix summarizes classification outcomes and errors. IoU measures the overlap between predicted and ground-truth bounding boxes, where higher values indicate better localization accuracy. Collectively, these metrics provide a comprehensive assessment of the YOLOv12m model's detection and classification performance.

E. Insightface face Recognition Model

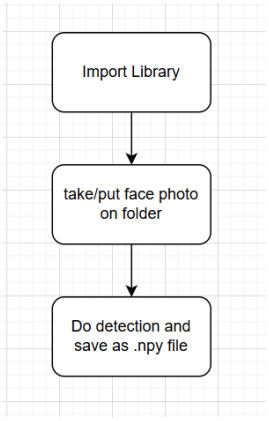

Fig 4. Face Recognition

This study employs a template-based face recognition method based on deep feature embedding matching. A pretrained ArcFace model with a ResNet-100 backbone is used as a fixed feature extractor to generate highly discriminative facial representations. The model was previously trained on large-scale face datasets using a deep metric learning strategy and remains unchanged during system operation.

Based on Fig.4 ,the face recognition pipeline consists of two main stages: enrollment and recognition. During enrollment, facial images are first detected and aligned to normalize pose variations. The aligned face images are then passed through the pretrained network to extract 512-dimensional embedding vectors, which are stored in .npy format as facial templates.

During the recognition stage, a query face image undergoes the same preprocessing and feature extraction process. The resulting embedding is compared with the stored templates using cosine similarity. The identity is determined based on the highest similarity score exceeding a predefined threshold. This matching-based approach allows the system to recognize identities accurately without requiring retraining or fine-tuning of the model.

III. RESULTS AND DICUSSION

A. *Model Performance and Results*

Based on the experimental results obtained after 692 training epochs, the YOLOv12m model demonstrated strong detection performance and stable convergence. The training process required a total time of 2329 seconds, indicating efficient utilization of computational resources throughout the training phase.

TABLE 1 RESULTS

Model	Precision	Recall	mAP50	F1-Score
YOLOv12m	99.38%	99.39%	99.5%	99.38%

In terms of detection accuracy, the model achieved a Precision of 99.38% and a Recall of 99.39%, demonstrating that YOLOv12m can reliably identify objects while maintaining both low false-positive and false-negative rates. The high mAP@0.5 value of 99.50% indicates excellent detection performance at the standard IoU threshold, while the mAP@0.5–0.95 of 98.76% reflects strong generalization across more stringent localization criteria.

The final training loss values—box loss of 0.2108, classification loss of 0.1770, and distribution focal loss (DFL) of 0.8176—indicate effective optimization of both localization and classification components. Correspondingly, the validation losses (box loss: 0.2263, classification loss: 0.1621, and DFL loss: 0.8111) remain close to the training losses, suggesting minimal overfitting and strong generalization to unseen data.

The learning rate for all parameter groups converged to approximately 2.77×10^{-4}, indicating that the optimizer reached a stable convergence point toward the end of training. Overall, these results confirm that YOLOv12m achieves a strong balance between detection accuracy, localization precision, and training stability, making it well-suited for real-world object detection applications.

B. *Confussion Matrix*

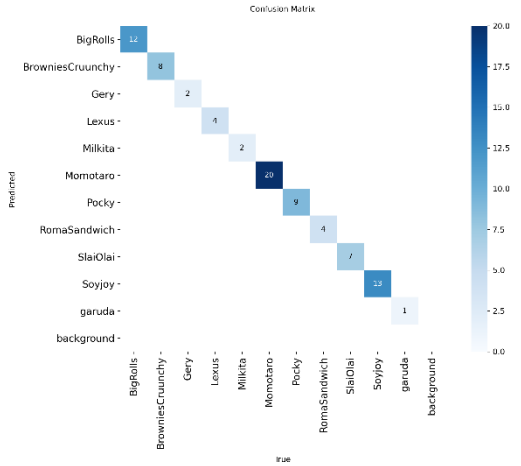

Fig 5. Confussion Matrix YOLOv12m

The confusion matrix in Figure 5 illustrates that the YOLOv12m model demonstrates strong classification performance across most product categories in the *jajan* dataset, as evidenced by the dominant diagonal values. Several classes, including Momotaro (20 correct predictions), Soyjoy (13), BigRolls (12), and Pocky (9), exhibit high true positive counts, indicating that the model successfully learns distinctive visual features for these products.

Misclassifications are minimal, with very few off-diagonal values observed, suggesting that inter-class confusion is limited. This indicates that most product classes possess sufficiently distinct visual characteristics, enabling the model to differentiate them accurately. The background class shows almost no incorrect predictions, implying that the model is effective at separating foreground objects from the background and produces a low false-positive rate.

However, several classes such as Gery (2 correct predictions), Milkita (2), Lexus (4), and Garuda (1) display lower correct classification counts. This is likely influenced by class imbalance and limited sample availability, rather than model instability. Despite this, no severe confusion between these classes and other product categories is observed, indicating that the model still maintains consistent decision boundaries.

Overall, the confusion matrix confirms that YOLOv12m exhibits stable and reliable classification behavior, with the majority of predictions correctly aligned along the diagonal. The observed errors are minor and primarily attributable to limited data representation rather than model deficiency.

.

C. *Training dan Validation*

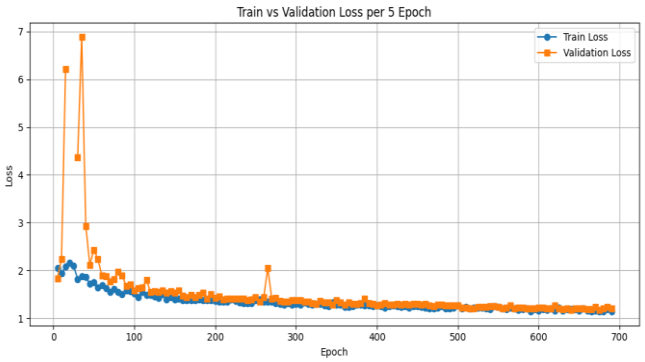

Fig 6. Train and Validation Loss YOLOv12m Graph

Figure 6 shows a consistent decline in both Training Loss and Validation Loss throughout the training process. The Training Loss decreases from approximately 6.0-3.0 in the early epochs to around 1.1 at epoch 692, while the Validation Loss drops sharply from about 6.7 to approximately 1.3 over the same period. After the initial stabilization phase, both curves decrease gradually and remain closely aligned, indicating effective learning and stable convergence. The absence of a widening gap between the two losses suggests that the YOLOv12m model does not suffer from overfitting and generalizes well to unseen data.

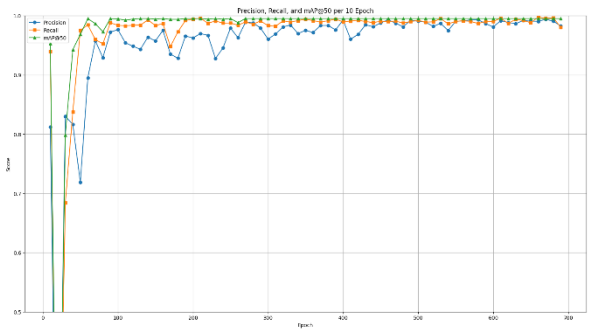

Fig 7. Precision, Recall, and mAP YOLOv12m Graph

The graph in Figure 7 illustrates a challenging training process with notable fluctuations in model performance. Precision starts very low at approximately 0.00 in epoch 1 and increases rapidly to around 0.86-0.98 by epoch 4-6, but then experiences significant instability between epochs 10-30, dropping to near 0.00 at epochs 19-20 before recovering to approximately 0.76-0.90 in later epochs. Recall follows a similar volatile pattern, rising sharply from 0.26 to above 0.93 within the first 10 epochs, then dropping dramatically to 0.00 during epochs 19-20, and gradually recovering to the 0.60-0.98 range by epoch 40-49. Meanwhile, mAP@50 begins at 0.05, increases to 0.95-0.98 by epochs 4-10, then drops to 0.00 during the unstable period (epochs 18-20), and stabilizes around 0.52-0.98 in the later training phases. The presence of NaN values in validation losses during epochs 18-29 suggests data loading or validation issues that contributed to the performance instability. Overall, while the model shows potential with high metrics when stable, the significant fluctuations and recovery periods indicate training instability that may require further optimization of hyperparameters, data augmentation, or learning rate scheduling.

D. Real-Time Model Testing with Camera

After the YOLO-based product detection model and the InsightFace face recognition model were successfully trained and their final weights were obtained, the research proceeded to the system implementation and real-time performance evaluation stage. This phase was conducted to validate the practical applicability of the proposed Automatic Retail Checkout System beyond offline evaluation metrics. The main objective was to evaluate the system's ability to accurately detect retail products and recognize customer identities simultaneously in a real-world, real-time environment.

The real-time testing was implemented using a Python-based application that employed the OpenCV library to capture continuous video streams from a camera. Each video frame was processed by the trained YOLO model to detect and classify products present at the checkout area, while the detected face regions were further analyzed using the InsightFace model for customer recognition. The system displayed bounding boxes, class labels, confidence scores, and recognition results in real time, enabling direct observation of detection accuracy, inference speed, and system responsiveness. This real-time evaluation demonstrates the feasibility of integrating YOLO-based product detection with face recognition for an efficient and automated retail checkout process.

Fig 8. Implementation jajan Model Testing

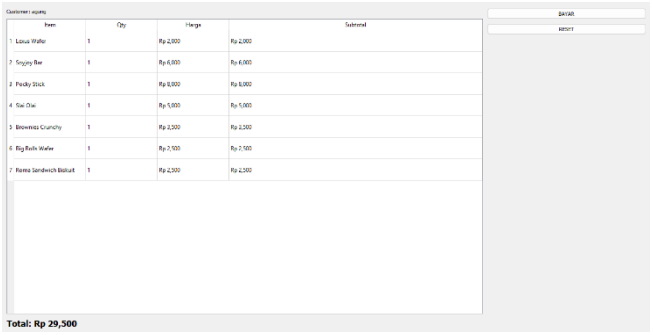

Fig 9. Jajan and customer username page

The system was operated by capturing input from a laptop camera, where each video frame was processed in real time by the trained YOLO-based product detection model. The implementation was executed by running the *mainvision.py* script within a preconfigured virtual environment. Using the OpenCV library, continuous video frames were acquired and forwarded to the model for product detection and classification.

As illustrated in Figure 8 and in Figure 9, the system successfully detected and classified multiple retail products placed within the checkout area and successfully detect face match with the username Each detected item is enclosed by a colored bounding box and labeled with its corresponding product name, unique ID, and confidence score. Higher confidence scores indicate greater certainty in the model's predictions, demonstrating reliable recognition of products such as snacks and packaged goods in a single frame.

Several aspects were observed to evaluate the system's real-time performance:

1. Inference Speed – The system achieved an average processing speed of approximately 15-20 frames per second (FPS) on the specified hardware, indicating suitability for real-time retail checkout scenarios.
2. Detection Accuracy – The model accurately detected and classified multiple products simultaneously, particularly when items were clearly visible and well-arranged within the camera's field of view.
3. Constraints and Challenges – Detection accuracy may decrease when products are partially occluded, overlapping, or affected by uneven lighting conditions, which can lead to lower confidence scores or misclassification.

Overall, the results demonstrate that the proposed YOLO-based product detection and face recognition system is effective for real-time automatic retail checkout applications. The system shows strong potential to reduce manual scanning processes; however, further improvements such as expanding the training dataset, & enhancing robustness to occlusion.

IV. CONCLUSION

Based on the research conducted, the proposed YOLOv12m-based product detection model demonstrates high detection performance, achieving a mAP@0.5 of 99.50% and a mAP@0.5–0.95 of 98.76%. The model also attained a Precision of 99.38% and a Recall of 99.39%, indicating reliable product identification with a low false-negative rate. Analysis of detection results shows that most products were correctly recognized, with minor misclassifications occurring mainly among visually similar items or partially occluded products. Furthermore, real-time testing confirmed stable system performance at an average processing speed of approximately 15 FPS, demonstrating its suitability for practical automatic retail checkout applications with minimal latency.

Despite these promising results, several limitations should be acknowledged. First, variations in lighting conditions, product occlusion, and cluttered backgrounds were found to negatively affect detection accuracy, as the model's performance remains sensitive to environmental factors commonly present in real retail environments. Second, although the dataset was sufficiently representative, it was collected under relatively controlled conditions and may not fully capture the diversity of product arrangements, packaging variations, and real-world usage scenarios. In addition, the current system focuses primarily on product detection and does not yet fully address challenges related to customer behavior, such as rapid object movement or complex interactions at the checkout area.

Future research should aim to address these limitations by expanding the dataset with more diverse retail conditions, applying more extensive data augmentation techniques, and enhancing robustness against occlusion and lighting variations. Furthermore, integrating more advanced customer identification and behavior analysis modules, such as improved face recognition and transaction tracking, could further enhance the system's reliability and applicability.

Overall, the proposed YOLO-based automatic retail checkout system shows strong potential for real-world deployment, offering an efficient and scalable solution to reduce manual checkout processes, improve transaction speed, and enhance customer experience in modern retail environments.

V. ACKNOWLEDGEMENTS

The authors would like to express their sincere gratitude to Universitas Muhammadiyah Yogyakarta (UMY) for the support and facilities provided, which greatly contributed to the completion of this research.

REFERENCE

[1] Dr. Sheetal janthakal, N Shivamani, Naveena A K, V Shrinivasa, "Auto checkout using yolo," International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), DOI: 10.17148/IJARCCE.2026.15146

[2] R. Skinderowicz, "Solving automatic check-out with fine-tuned YOLO models," *Procedia Computer Science*, vol. 246, pp. 1649–1658, 2024, doi: https://doi.org/10.1016/j.procs.2024.09.644.

[3] L. Tan, S. Liu, J. Gao, X. Liu, L. Chu, and H. Jiang, "Enhanced Self-Checkout System for Retail Based on Improved YOLOv10," *Journal of Imaging*, vol. 10, no. 10, p. 248, Oct. 2024, doi: https://doi.org/10.3390/jimaging10100248.

[4] N. James, "Automated Checkout for Stores: A Computer Vision Approach," *Revista Gestão Inovação e Tecnologias*, vol. 11, no. 3, pp. 1830–1841, Jun. 2021, doi: https://doi.org/10.47059/revistageintec.v11i3.2053.

[5] M. Ariyanto and Prima Dewi Purnamasari, "Object Detection System for Self-Checkout Cashier System Based on Faster Region-Based Convolution Neural Network and YOLO9000," Oct. 2021, doi: https://doi.org/10.1109/qir54354.2021.9716200.

[6] P. P. Patil, Ankit Naresh Bhoir, Sayali Santosh Tembe, and Krupa Kishor Madrewar, "Smartcart Vision," *Zenodo (CERN European Organization for Nuclear Research)*, Jun. 2025, doi: https://doi.org/10.5281/zenodo.18104842.

[7] C. Zarco, J. Giráldez-Cru, O. Cordón, and F. Liébana-Cabanillas, "A comprehensive view of biometric payment in retailing: A complete study from user to expert," *Journal of Retailing and Consumer Services*, vol. 79, p. 103789, Jul. 2024, doi: https://doi.org/10.1016/j.jretconser.2024.103789.

[8] S. Wang, G. Mortimer, L. Sajtos, and B. Keating, "Exploring consumers' competence, autonomy and relatedness needs in the adoption of facial recognition payment technology," *Journal of Retailing and Consumer Services*, vol. 81, p. 104044, Aug. 2024, doi: https://doi.org/10.1016/j.jretconser.2024.104044.

[9] D. Nan, Y. Kim, J. Huang, H. S. Jung, and J. H. Kim, "Factors Affecting Intention of Consumers in Using Face Recognition Payment in Offline Markets: An Acceptance Model for Future Payment Service," *Frontiers in Psychology*, vol. 13, Mar. 2022, doi: https://doi.org/10.3389/fpsyg.2022.830152.

[10] Y. Zhou, N. Wu, B. Hu, Y. Zhang, J. Qiu, and W. Cai, "Implementation and Performance of Face Recognition Payment System Securely Encrypted by SM4 Algorithm," *Information*, vol. 13, no. 7, p. 316, Jun. 2022, doi: https://doi.org/10.3390/info13070316.

[11] R. Rija, R. Muttasher, G. Al-Araji, and A. Ghaida, "Payment Systems Based on Face Recognition: A Survey," *Guangdianzi Jiguang/Journal of Optoelectronics Laser*, vol. 41, pp. 563–571, 2022.

[12] I. Dijmărescu, M. Iatagan, I. Hurloiu, M. Geamănu, C. Rusescu, and A. Dijmărescu, "Neuromanagement decision making in facial recognition biometric authentication as a mobile payment technology in retail, restaurant, and hotel business models," *Oeconomia Copernicana*, vol. 13, no. 1, pp. 225–250, Mar. 2022, doi: https://doi.org/10.24136/oc.2022.007.