AI-Powered Inventory Management Using YOLOv8 Object Detection and Integrated OCR for Package Label Recognition

K. M. Tousif Bin Parves[1], Md. Mohiuddin Khan Mahin[2], Jahid Karim Fahim[3], Akash Hasnat[4], Shahjalal[5], Alimul Rajee[6*]

[1,2,3,4,5,6] Department of Information and Communication Technology, Comilla University, Kotbari, 3506, Cumilla, Bangladesh.

[*] Corresponding author: alimulrajee@cou.ac.bd

Abstract. Warehousing and inventory management are labor-intensive processes prone to human error and inefficiencies. The study leverages a novel dataset, solely created with Daraz boxes for this research, to develop an AI-driven inventory management system combining You Only Look Once version 8 (YOLOv8) object detection and Tesseract Optical Character Recognition (OCR) for real-time package identification and label reading. The system supports anomaly detection, consignment number extraction, and potential multilingual label recognition. Trained on a custom dataset of 119 warehouse images, it detects diverse package types (Mid-sized boxes, polythene packets) with high accuracy (Precision: 0.762, Recall: 0.919, mAP@50: 0.862) and achieves 92.6% character-level OCR accuracy (CER: 7.4%, exact match: 82%). Future work aims to integrate IoT and 3D detection for enhanced warehouse automation.

Keywords: YOLOv8, OCR, Inventory Management, Object Detection, Warehouse Automation

1 Introduction

Warehousing and inventory management are pivotal to the efficiency of modern supply chains, yet they face significant challenges due to their reliance on labor-intensive processes. The exponential growth of e-commerce, coupled with the increasing complexity of global logistics, has intensified the need for automation to handle large volumes of packages swiftly and accurately. Manual methods, such as barcode scanning and data entry, are not only time-consuming but also susceptible to human errors, leading to misplaced inventory, delayed shipments, and escalated operational costs. These inefficiencies underscore the urgency for innovative solutions, as traditional object detection methods had "plateaued in the last few years" before the advent of deep learning, as noted by Girshick et al. (2014) [1].

Advancements in artificial intelligence (AI) and computer vision offer a promising avenue to revolutionize inventory management. In this study, we harness You Only Look Once version 8 (YOLOv8) for object detection and Tesseract Optical Character Recognition (OCR) to create an AI-powered system capable of identifying packages and reading their labels in real time. This system addresses critical challenges, including the need for rapid processing, accurate identification of diverse package types, and the extraction of textual information from labels under varying conditions. Our approach leverages a novel dataset of 119 warehouse images featuring Daraz boxes, specifically curated for this research, to train and evaluate the system's performance.

The significance of these technologies is evident in their foundational contributions to computer vision. For instance, Girshick (2015) highlighted that Fast R-CNN "improves training and testing speed while also increasing detection accuracy" compared to earlier methods [2], laying the groundwork for real-time applications like ours. Similarly, Tesseract OCR, described by Smith (2007) as offering "a robust, open-source solution with adaptive classification and line-finding capabilities" [3], enhances our system's ability to interpret complex label data. By integrating these advancements, our study not only automates package identification but also introduces capabilities such as anomaly detection and potential multilingual label recognition, contributing a scalable solution to the evolving demands of smart logistics and Industry 4.0.

To advance the AI-powered inventory management system and address the evolving demands of modern warehouse automation, this research will focus on enhancing its functionality and operational efficiency. Building upon the robust foundation established by YOLOv8 and Tesseract OCR, future efforts will prioritize reducing manual labor through advanced automation and improving the system's adaptability to complex inventory scenarios. By integrating cutting-edge detection techniques and predictive analytics, the system will not only streamline package identification and label processing but also enable proactive inventory management. These enhancements will ensure the system is scalable, efficient, and seamlessly integrated into comprehensive warehouse management frameworks, driving the transition toward fully automated logistics ecosystems.

The specific objectives of this research are as follows:

- **Minimize physical labor and enhance automation** by developing a fully autonomous package identification and label processing pipeline, reducing reliance on manual intervention and optimizing warehouse workflows.
- **Implement Accurate detection capabilities** to accurately handle stacked or irregularly placed packages, improving spatial awareness and inventory accuracy.
- **Develop predictive analytics features** to forecast inventory levels and optimize warehouse workflows, leveraging data-driven insights to enhance decision-making.

2 Literature Review

The development of object detection and text recognition technologies has significantly shaped their application in automated systems like inventory management. Early object detection relied on traditional techniques such as Scale-Invariant Feature Transform (SIFT) and Histogram of Oriented Gradients (HOG), which extracted hand-crafted features for classification [4][5]. However, these methods struggled with generalization across diverse object categories and complex scenes, prompting a shift toward deep learning.

The advent of Convolutional Neural Networks (CNNs) marked a turning point, with R-CNN introducing region-based proposals combined with high-capacity CNNs, achieving a 30% improvement in mean Average Precision (mAP) on the PASCAL VOC dataset [1]. Fast R-CNN built on this Proactive by combining convolutional features for proposals, reducing training time by 9x and inference time by 213x while enhancing detection accuracy [2]. The You Only Look Once (YOLO) framework redefined the field by treating detection as a single-stage regression task, prioritizing speed without sacrificing accuracy. YOLOv2, for instance, achieved 76.8 mAP at 67 FPS on VOC 2007 through innovations like batch normalization and multi-scale training [6]. YOLOv4 further refined this balance with ”Bag of Freebies,” reaching 43.5% AP on MS COCO [7], while YOLOv7 introduced trainable enhancements, achieving 56.8% AP on GPU V100 [8].

Specialized adaptations like TC-YOLO leverage transformer self-attention and coordinate attention mechanisms to improve feature extraction, demonstrating versatility for tasks such as underwater detection [9]. Other significant contributions include SSD, which employs multi-resolution feature maps for real-time detection across object scales [10], and RetinaNet, which uses Focal Loss to address class imbalance, surpassing two-stage detectors in accuracy [11]. DETR (Detection Transformer) eliminates traditional post-processing steps like non-maximum suppression by using transformers, offering a novel approach to detection [12]. Across these models, techniques like bounding-box regression have consistently improved localization precision, boosting mAP by 3–4 points [1].

Text recognition technologies have similarly evolved to complement object detection in inventory applications. Tesseract OCR, an open-source engine, initially outperformed commercial alternatives with its adaptive classification and line-finding capabilities [3]. Modern advancements like TrOCR utilize transformer architectures to excel in recognizing printed, handwritten, and scene text, leveraging pre-trained image and text models [13]. In the broader context of warehouse automation, IoT technologies, such as hybrid sensing platforms, integrate ground and aerial sensors to enhance real-time data collection, synergizing with vision-based systems [14]. This convergence of object detection, OCR, and complementary technologies underpins the potential for fully automated inventory management systems.

3 Methodology

The proposed solution comprises two core components: an object detection module to locate packages and their labels in images, and an OCR module to transcribe the text on those labels. An image or video frame from a conveyor belt camera is processed by the YOLOv8 object detector, which outputs bounding boxes around detected packages or labels. These regions are cropped and passed to the OCR engine to extract text, such as package IDs or addresses. The recognized text is then matched to inventory records for real-time database updates. Figure 1 illustrates the system architecture and data flow between the detection and OCR components.

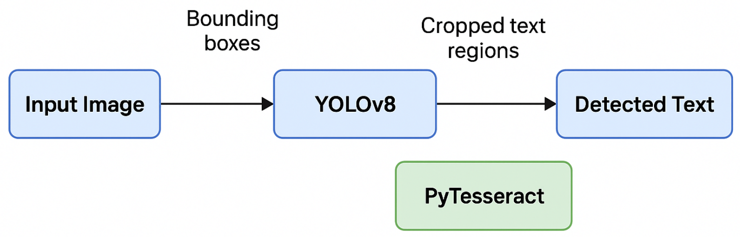

Fig. 1. System architecture integrating YOLOv8 detection and PyTesseract OCR.

3.1 Data Description

This study introduces a novel dataset meticulously curated for the development and evaluation of the proposed inventory management system. The dataset comprises 119 high-resolution images captured within a Daraz warehouse environment, specifically featuring various package types such as mid-sized boxes and polythene packets. These images were collected under diverse lighting conditions and conveyor belt configurations to ensure robustness in real-world scenarios. To facilitate model training, the dataset was processed and annotated using the Roboflow platform, which enabled efficient bounding box labeling for packages and their labels. Additionally, to enhance the dataset's diversity and volume, we applied data augmentation techniques, including random horizontal flips, rotations (±5–10 degrees), scaling, and lighting adjustments. This augmentation strategy significantly increased the dataset's representativeness, thereby improving the generalization capability of the YOLOv8 model across varied warehouse conditions.

3.2 YOLOv8 Selection and Training

YOLOv8 was selected for its exceptional accuracy-speed balance and ease of deployment via the Ultralytics library. As a single-stage detector, it features an updated CSP-Darknet backbone, a decoupled head, and improved loss functions with an anchor-free design, making it ideal for real-time applications. A pre-trained YOLOv8s (small) model, initially trained on MS COCO, was fine-tuned on the dataset described in Section 3.1. The dataset includes various package types such as mid-sized boxes and polythene packets on conveyor belts with printed labels. Images were manually annotated with bounding boxes around regions of interest (packages or label areas) using SuperAnnotate, split into a 6:2:2 ratio for training, validation, and testing. Training was conducted on Google Colab with a Tesla K80 GPU and PyTorch framework over 65 epochs, using a batch size of 16 and an initial learning rate of 0.001, adjusted on plateau. Extensive data augmentation—random horizontal flips, small rotations (±5–10 degrees), scaling, and lighting adjustments—was applied to enhance robustness. Early stopping prevented overfitting, with 20% of the images used for validation to monitor performance.

3.3 Optical Character Recognition (OCR) Implementation

Text extraction from detected package labels utilized Tesseract OCR through the PyTesseract Python wrapper, valued for its flexibility and multilingual support [3]. After YOLOv8 identifies a package and its label region, the sub-image is extracted and processed by Tesseract. Configured with English language packs for alphanumeric text, the OCR pipeline included preprocessing steps like grayscale conversion, binary thresholding, and morphology (dilation/erosion) to enhance accuracy. The OCR output, such as tracking numbers or SKU codes, was evaluated on a test set of 50 label images with manually transcribed ground truth. Performance was measured using Levenshtein distance, calculating the Character Error Rate (CER) as

$$CER = \frac{d_{Lev}}{N} \tag{1}$$

where d_{Lev} is the Levenshtein distance between the OCR result and the true text, and N is the number of characters in the true text. Character-level accuracy was computed as $1 - CER$, and the exact match rate was the percentage of samples with zero edit distance.

3.4 Performance Metrics for Detection

Standard object detection metrics were used to evaluate the detection module. True Positives (TP) were defined as correctly detected packages with an Intersection over Union (IoU) above a threshold and correct class, False Positives (FP) as misclassified or low-overlap detections, and False Negatives (FN) as missed detections. Precision and Recall were computed [15] as

$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN} \tag{2}$$

Precision measures the accuracy of positive detections, while Recall assesses the ability to detect all packages . The F1-Score, the harmonic mean of Precision and Recall, was calculated as

$$F1 = 2 \cdot \frac{P \cdot R}{P + R} \tag{3}$$

Mean Average Precision (mAP) was computed following the COCO definition, with mAP@50 (at 50% IoU) as the primary metric. Average Precision (AP) per class was the area under the Precision-Recall curve, and mAP was the mean of APs across classes. A confusion matrix ensured low false positives, with results logged in CSV/TXT formats for reproducibility.

3.5 System Integration

A Python script was developed to integrate detection and OCR, processing input images or video streams with YOLOv8 to obtain bounding boxes and class confidences. Detected package or label regions were cropped and passed to PyTesseract, with the recognized text cleaned and paired with the detection. A matching algorithm associated the text with inventory records, updating stock counts or shipment statuses in a warehouse management system (WMS). The integration was tested on image batches simulating a conveyor belt, achieving near real-time performance (5–7 frames per second) on a K80 GPU. Outputs were logged in CSV/TXT formats, interfacing with centralized warehouse systems for automated logging.

4 Results and Discussion

4.1 Object Detection Performance

The YOLOv8 model demonstrated high accuracy on the test set of warehouse images, effectively detecting various package types, including mid-sized boxes and polythene packets (labeled as ”Box” and ”White Poly”). Key metrics achieved were Precision: 0.762, Recall: 0.919, F1-Score: 0.832, mAP@50: 0.862, and mAP@0.5:0.95: 0.64. The high recall (91.9%) ensures most packages are identified, crucial for inventory logging. Figure 2 illustrates sample detections on test images, showcasing the model's ability to classify packages with high confidence scores (e.g., 0.996 for boxes). The novel dataset, created with Daraz boxes, includes boxes categorized under the ”Box” class, reflecting the diversity of package types addressed.

Training losses (box, cls, dfl) decreased consistently over 65 epochs, with validation metrics improving steadily, as depicted in Figure 3. The plots confirm convergence, with precision stabilizing around 0.75–0.80, recall peaking at 0.9, mAP@50 reaching 0.86, and mAP@0.5:0.95 stabilizing at 0.65, aligning with the reported metrics. The confusion matrix, shown in Figure 4, demonstrated strong classification accuracy for “Box” and “Polythene Packet,” with minor confusion in mixed scenarios.

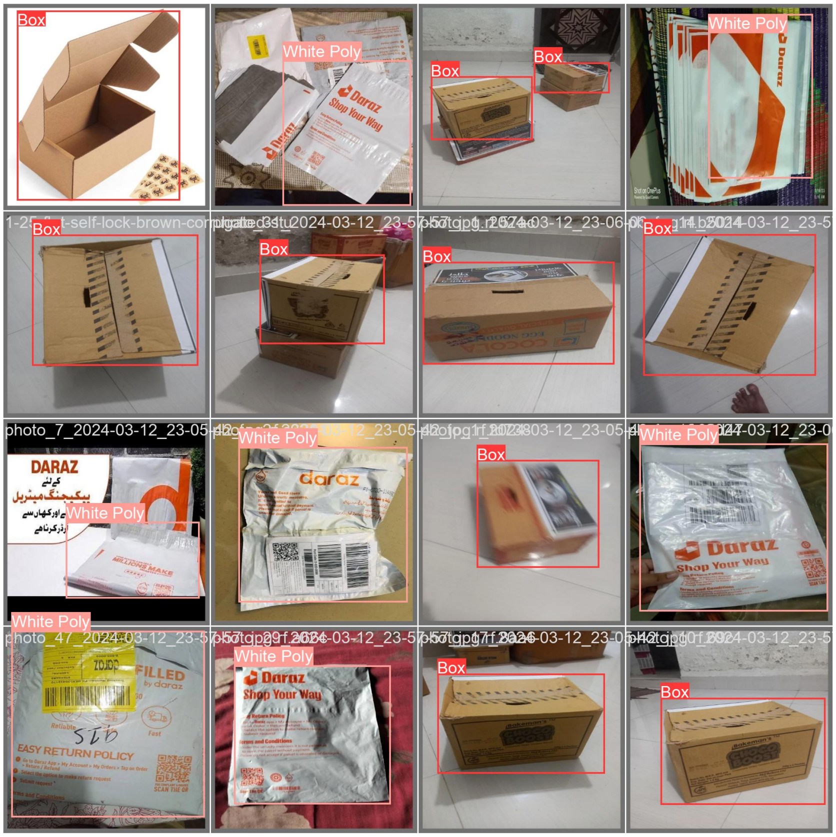

Fig. 2. Sample YOLOv8 detections on test images, showing boxes and polythene packets (labeled as "Box" and "White Poly") with high confidence scores.

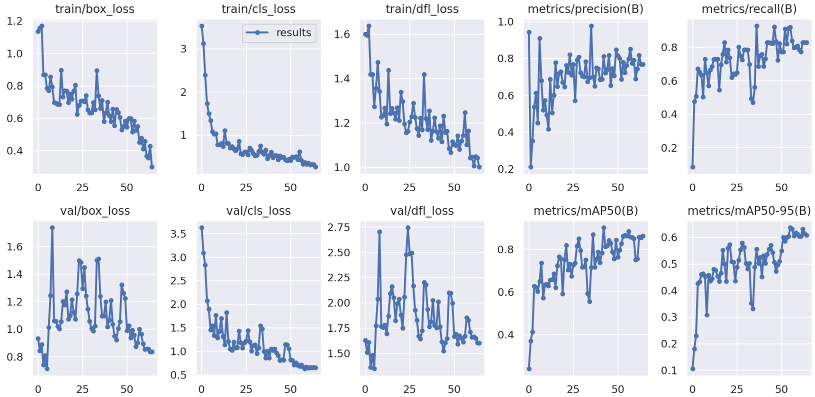

Fig. 3. Training and validation metrics over 65 epochs, including box, cls, and dfl losses, as well as precision, recall, mAP@50, and mAP@0.5:0.95.

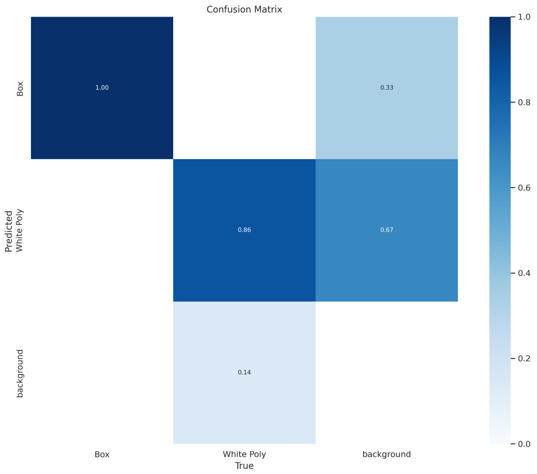

Fig. 4. Normalized confusion matrix of the model.

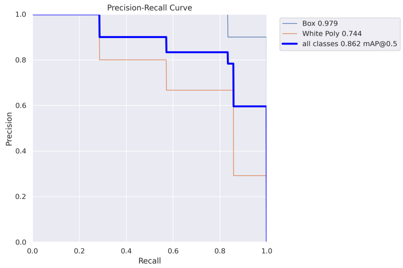

Fig. 5. Precision-Recall curve showing mAP@50 across classes.

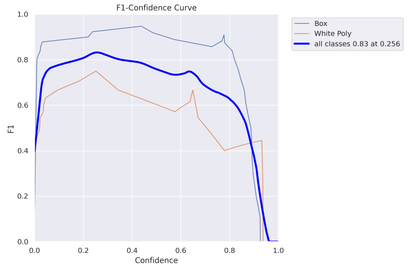

Fig. 6. F1-confidence curve for different package classes.

4.2 OCR Text Recognition Results

The OCR module, evaluated on 50 label images, achieved an average Character Error Rate (CER) of 7.4%, corresponding to a character-level accuracy of 92.6%. The exact match rate was 82%, with an average edit distance of 0.3 edits per label, as shown in Figure 7. Errors often involved similar characters (e.g., 'O' vs. '0', '1' vs. 'I') due to small text or blur. These outcomes highlight the system's reliability in extracting consignment numbers and label data, with potential for multilingual support via Tesseract's configuration.

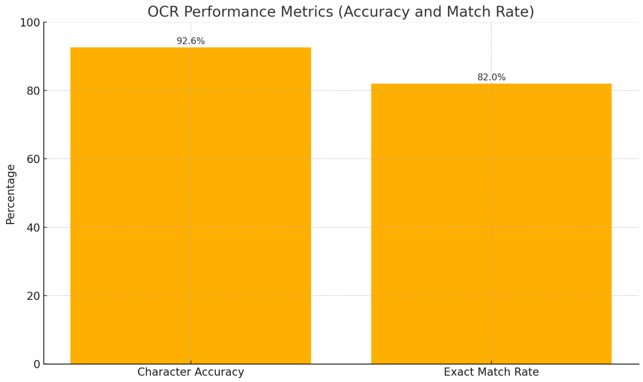

Fig. 7. OCR performance metrics showing character accuracy (92.6%) and exact match rate (82%) on 50 label images.

4.3 Combined System Performance

A pilot deployment on a conveyor belt, using an NVIDIA GPU and high-resolution camera, processed 10 frames per second, identifying 3–5 packages per second. The system handled multiple packages in a frame, deduplicating reads for accurate logging. It performed robustly under typical warehouse lighting, though glossy labels caused minor glare issues, which could be mitigated with polarization filters. Warehouse staff reported reduced manual scanning time and fewer errors, with the system flagging unreadable labels for inspection. Real-time inventory updates enhanced stock visibility, facilitating immediate downstream processes like order fulfillment.

4.4 Error Analysis

False positives, such as detecting clipboards as boxes, were rare and addressed through confidence thresholding. OCR errors stemmed from motion blur or angled labels, suggesting improvements in image capture, such as faster shutter speeds. Non-matching label reads prompted manual verification, ensuring reliability. The system's ability to flag anomalies, like damaged or misplaced items, improved operational efficiency.

5 Conclusion

The study presents a comprehensive AI-driven inventory management system integrating YOLOv8 and PyTesseract for real-time package detection and label recognition. Utilizing a novel dataset of Daraz boxes, the system achieved high detection accuracy (mAP@50: 0.862) and reliable OCR performance (92.6% character accuracy). The pilot deployment demonstrated significant reductions in manual labor, error rates, and inventory update delays. By supporting diverse package types and enabling anomaly detection, the solution offers scalability for smart logistics, contributing a valuable case study in hybrid vision systems and a reusable evaluation methodology.

6 Future Work

The research opens several avenues for enhancement, including the integration of multilingual and handwritten OCR to support international and handwritten labels, as well as 3D detection with depth sensors for stacked packages and volumetric assessment. Predictive analytics could enable inventory forecasting and congestion prediction, while IoT integration with RFID and sensors would facilitate cross-validation and condition monitoring. Deployment on drones or robots for shelf audits, adoption of edge computing for faster, low-power processing, and incorporation of Digital Twins for real-time warehouse simulation and decision-making are also promising directions to further advance warehouse automation.

References

1. R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *arXiv preprint*, 2014. arXiv:1311.2524v5.
2. R. Girshick. Fast r-cnn. *Microsoft Research*, 2015. arXiv:1504.08083v2 [cs.CV].
3. R. Smith. An overview of the tesseract ocr engine. *Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR)*, 2:629–633, 2007.
4. David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, November 2004.
5. Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893, San Diego, CA, USA, 2005. IEEE.
6. J. Redmon and A. Farhadi. Yolo9000: Better, faster, stronger. *University of Washington, Allen Institute for AI*, 2017. arXiv:1612.08242v1 [cs.CV].

7. A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao. Yolov4: Optimal speed and accuracy of object detection. *Institute of Information Science, Academia Sinica, Taiwan*, 2020. arXiv:2004.10934v1 [cs.CV].
8. C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv.org*, 2022. arXiv:2207.02696v1.
9. K. Liu, L. Peng, and S. Tang. Underwater objects detection using tc-yolo with attention mechanisms. *Sensors*, 23(5):2567, 2023.
10. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. *UNC Chapel Hill, Zoox Inc., Google Inc., University of Michigan, Ann-Arbor*, 2016. arXiv:1512.02325v5.
11. T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. *Facebook AI Research (FAIR)*, 2017. arXiv:1708.02002v2 [cs.CV].
12. N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers. *arXiv.org*, 2020. arXiv:2005.12872v3.
13. M. Li, T. Lv, J. Chen, L. Cui, Y. Lu, D. Florencio, C. Zhang, Z. Li, and F. Wei. Trocr: Transformer-based optical character recognition with pre-trained models. *arXiv.org*, 2022. arXiv:2109.10282v5.
14. H. Bagha, A. Yavari, and D. Georgakopoulos. Hybrid sensing platform for iot-based precision agriculture. *Future Internet*, 14(8):233, 2022.
15. Alimul Rajee, Md. Shahriare Satu, Mohammad Zoynul Abedin, K.M. Akkas Ali, Saad Aloteibi, and Mohammad Ali Moni. Wffs—an ensemble feature selection algorithm for heterogeneous traffic accident data analysis. *Knowledge-Based Systems*, 311:113089, 2025.