A Machine Learning-Based Detection and Recognition Approach for 1D Barcode Scanning Through Visible Light Communication

Ruchi Patel
Independent Researcher

Prit Patel
Independent Researcher

Abstract—A dependable and affordable method of product marking is the barcoding system. For an inventory system to identify product barcodes and for the product billing system, the barcode detection procedure is required. Barcode scanning effectiveness combined with accuracy proves crucial for all retail operations and logistics management along with inventory control systems. Traditionally barcode recognition needs dedicated hardware instruments that prove costly and imposing. The research investigates how a machine learning detection and recognition system for 1D barcode identification functions with optimized advanced models for superior accuracy and efficiency. The proposed research develops an advanced 1D barcode detection system by utilizing the state-of-the-art YOLOv3 object detection model. YOLOv3 operates as a machine learning model that optimizes confidence scores and includes Darknet-53 and multi-scale detection with residual blocks to enhance barcode recognition capabilities. Experimental trials show that the proposed technique reaches Jaccard accuracy of 93.9% (Javg) alongside a detection rate of 99.8% (D0.5) exceeding YOLOv2, Least Square, and MSER. The model maintains accurate barcode detection in challenging situations which include obscured images and distortions and light reflection indicators while offering real-time capabilities. YOLORv3 delivers remarkable barcode detection capabilities which position it as a dependable tool for industrial use in commercial settings. The developed method builds superior barcode recognition capabilities through its real-time scanning system which performs effectively across multiple environments.

Keywords—Barcode technology, 1D barcode scanning, barcode detection, Visible Light Communication (VLC), machine learning (ML), WWU Muenster Barcode Data.

I. Introduction

Barcodes have rapidly developed and expanded around the world because of their inexpensive manufacturing and secure handling. Barcodes were the last addition to the labelling of various products when the laser scanner was developed in the 1970s. Due to the widespread availability of low-cost cameras, laser-based barcode scanners are increasingly being replaced by them [1]. It became necessary to develop algorithms for barcode localization and orientation estimation when cameras were used as barcode capture devices. The usage of mobile devices, such as cell phone cameras with their tiny imaging sensors, made the issues of sensor noise, motion blur, and dim illumination even more noticeable.

The issue of random barcode alignment in the picture and complex environmental conditions are challenges for camera-based barcode decoders. A number of visual distortions, including perspective warp and spatial occlusion, may be introduced by the haphazard alignment of barcodes. Different levels of blur, overexposure, and brilliant patches of light reflection can also be generated by environmental variables, such as shooting distance, camera settings, and light conditions. A good barcode detector must be able to overcome these environmental obstacles.

There are uses for 1D barcode detection in industries including manufacturing, retail, logistics, and transportation. In addition to the conventional human-operated handheld laser barcode reader, vision-based barcode scanning has garnered significant interest recently due to its ability to automatically identify objects, hence enabling a greater level of automation [2]. The goal of the reliable automated 1D barcode detection method is to use sensor vision to identify a pure 1D barcode region and rotation angle.

The process of efficient 1D barcode scanning requires state-of-the-art image processing along with segmentation methods [3]. The automatic detection of barcodes under different lighting conditions and various distortions and blocked areas depends on both barcode region detection accuracy and rotational adjustment. VLC becomes more accurate at barcode segmentation when integrated with detection mechanisms that optimize the identification process. The recognition reliability requires sophisticated computational models in order to achieve maximum effectiveness.

The combination of AI alongside ML and DL used for 1D barcode recognition within VLC introduces a ground-breaking answer. The learning process of AI models examines barcode patterns and handles deformed codes to boost detection efficiency. ML improves scanning efficiency through negative result elimination and DL strengthens both barcode division processes and decoding mechanisms. The integration of AI, ML and DL techniques with VLC allows this research to create a secure system that increases barcode scanning reliability and efficiency for modern automated systems.

A. *Motivation and Contributions of the Study*

Multiple business sectors, especially healthcare, along with logistics and retail, depend on barcode scanning for vital product identification services. Traditional barcode detection systems often struggle with challenges such as poor lighting conditions, occlusions, distortions, and reflections, leading to inaccurate or failed scans. With the advancement of DL techniques, object detection models like YOLO have shown remarkable improvements in real-time detection tasks. The motivation behind In order to improve efficiency and automation in barcode-based processes, this project aims to create a high-performance, real-time barcode scanning system that can be successfully applied in a variety of real-world applications. This article makes the following contributions:

- Uses the WWU Muenster Barcode Database as the main source of information for barcode identification and detection.
- Standardizes image resolution (640 × 480 pixels) and applies grayscale conversion, binarization (Otsu's method), and noise filtering to improve barcode visibility.
- Transforms images into grayscale (0–255 intensity levels) to enhance edge detection and contrast adjustment for barcode recognition.
- Uses Gaussian and median filtering to eliminate noise, improving barcode visibility and ensuring accurate detection.

- Implemented and design the YOLOv3 model for ID barcode scanning
- Measuring model effectiveness using evaluation metrics such as Jaccard Accuracy and detection rate to determine the most efficient approach.

B. Justification and Novelty

The justification for this study arises from the growing need for efficient and accurate barcode scanning methods, particularly in challenging lighting conditions. Traditional barcode detection relies on specialized hardware, limiting scalability and adaptability. By integrating Visible Light Communication (VLC) with machine learning, this research enhances barcode recognition using the WWU Muenster Barcode Database. The novelty lies in employing an optimized YOLOv3 model with advanced preprocessing and data augmentation techniques to improve detection accuracy. Unlike conventional methods, this approach leverages VLC's capabilities to enhance robustness, ensuring high-speed, real-time barcode recognition with improved adaptability in dynamic environments.

C. Organization of the paper

Here is the structure of the paper: Section II reviews related research on barcode scanning based on machine learning. Section III details data preprocessing, feature extraction, and evaluation metrics. Section IV compares model performance using visual analysis. Finally, Section V summarizes key findings and suggests improvements for VLC-based barcode scanning.

II. Literature Review

The literature review explores machine learning-based approaches for detecting and recognizing patterns in visual data transmitted through optical channels and exploring hybrid and ensemble techniques to optimize scanning efficiency and real-time processing capabilities. Some of reviews are:

Sharif et al. (2020) research suggests a combination barcode and text orientation detection approach based on the observation that 1D barcodes are typically aligned with text. Their method uses a CNN-based barcode localization network to first identify the four vertices of an arbitrarily aligned 1D barcode. After that, a post-processing module uses it to determine the text's alignment angle. Tests using a combination of publicly available and privately obtained data confirm that their method can precisely locate barcode areas and improve the OCR robustness in unattended retail systems [4].

Kalinov et al. (2020) enhances the UAV's localization in a real warehouse with low light levels by using scanned barcodes as landmarks. For flight-path optimization based on barcode locations, it employs a unique method in place of the conventional OSBG trajectory. This method shortens the time needed for warehouse inventory and lowers the likelihood of barcode scanning errors [5].

Burta, Szabo and Gontean (2020) the system compares an image to the images in the database to identify items. Every database picture has a string attached that describes the entity it depicts. When the recently captured image matches one that is already in the database, the new image uses the string from the matching image. Train the neural network in this manner. The system's accuracy increases with the number of photographs stored in the database. The system now supports voice commands and text-to-speech so that it can be a genuine robot that can be asked what an object is and then pronounce it aloud after the system has identified it [6].

Jia et al. (2019) to ascertain the locations and forms of the barcodes, the ROI is forwarded to the classification and regression module. Lastly, the geometric distortion is eliminated using the distortion removal module in accordance with the regression parameters obtained in the preceding stage. Their approach can identify and fix the distorted barcode form and exact location. In experiments, their approach on a difficult large-scale dataset. In contrast to other approaches, their approach offers a comprehensive solution for precisely locating barcode vertexes, demonstrating superior detection accuracy [7].

Zhang et al. (2019) the application scenario has become more complex as a result of the increasing usage of barcode technology. The issues of uneven light, distortion, and shelter are not universally resolved by the conventional barcode recognition approach. The DL theory is applied in this research to tackle the barcode identification problem in the aforementioned scenario. Accordingly, the issue of fixing linear distortion. The key technology for barcode identification in challenging situations has been developed, and the Data Matrix code has been solved. Following testing, the identification accuracy was around 93% and the speed was 125ms [8].

Ventsov and Podkolzina (2018) the authors progressively added several kinds of distortions to the preexisting training set of photos, matching the most prevalent kinds of faults, in order to learn and identify typical errors. The validation sample's feedback was used to determine how many "distorted" photos were added. The structure of a convolutional neural network for the barcode localization technique was created by the authors. This neural network achieved a 97.41% accurate barcode identification rate for source photos. The proper detection percentage was 85.68% for photos that were rotated and 83.32% for images that had brightness alterations. Similar methods showed improvements of 1.32%, 7.21%, and 10.68%, respectively [9].

Table I presents the research gaps in previous studies on ML-based approaches for detecting and recognizing 1D barcodes using Visible Light Communication. It highlights key limitations in existing methodologies, datasets, performance benchmarks, and real-world applicability, providing a foundation for further exploration and improvement.

TABLE I. SUMMARY OF THE MACHINE LEARNING-BASED 1D BARCODE DETECTION AND RECOGNITION

Author	Methodology	Dataset	Performance	Limitations & Future Work
Sharif et al. (2020)	A CNN-based localisation network is used for joint barcode and text orientation detection, and post-processing is used to estimate angle.	Combined public and self-collected datasets.	Accurate barcode localization and enhanced OCR robustness in unmanned retail systems.	Limited to OCR-focused applications; future work can expand to real-time applications and more complex environments.
Kalinov et al. (2020)	Barcode-aided UAV localization in low-light warehouses using a novel flight-path optimization over standard OSBG.	Real-world warehouse environment.	Improved localization and reduced scanning errors and time during stocktaking.	Domain-specific to UAVs and warehouses; could be extended to other robotic platforms and environments.
Burta, Szabo, and Gontean (2020)	Image matching with database entries; uses text-to-speech and voice commands for object recognition.	Internal image database with labeled strings.	Accuracy improves with more database images; supports interactive recognition.	Dependent on database size and quality; future enhancements may include real-time learning and barcode-specific adaptations.
Jia et al. (2019)	End-to-end detection pipeline with classification, regression, and distortion removal modules.	Challenging large-scale dataset.	High detection accuracy and precise shape/position correction of barcodes.	High computational load; future work could focus on lightweight architectures for mobile deployment.
Zhang et al. (2019)	Deep learning-based method to address illumination, distortion, and occlusion in barcode detection.	Not specified; tested in complex conditions.	125ms recognition speed with ~93% accuracy.	Dataset limitations not discussed; future work could involve more diverse scenarios and generalization capabilities.
Ventsov and Podkolzina (2018)	CNN trained with incremental addition of distorted images to improve robustness against rotation and brightness variation.	Custom dataset with systematically added distortions.	97.41% accuracy for source images, 85.68% for rotated, 83.32% for brightness-affected images.	Model improvement needed for extreme lighting variations; future work could integrate adaptive augmentation techniques.

III. METHODOLOGY

The research methodology for 1D Barcode Scanning Through VLC involves dataset selection, preprocessing, VLC implementation, model training, and evaluation. The WWU Muenster Barcode Database, which contains 1,055 barcode images with a resolution of 640 × 480 pixels, is used. After that pre-process the data includes image resizing, rotation, grayscale conversion, binarization (Otsu's method), and filtering (Gaussian/median filtering) to enhance barcode clarity. There are 20% for testing and 80% for training in the dataset. The YOLOv3 model is trained on preprocessed images for barcode detection. Then, the model is evaluated on the performance matrix, like Jaccard Accuracy (J(avg)) and Detection Rate ($D_{0.5}$). The methodology's general phases are displayed in Figure 1.

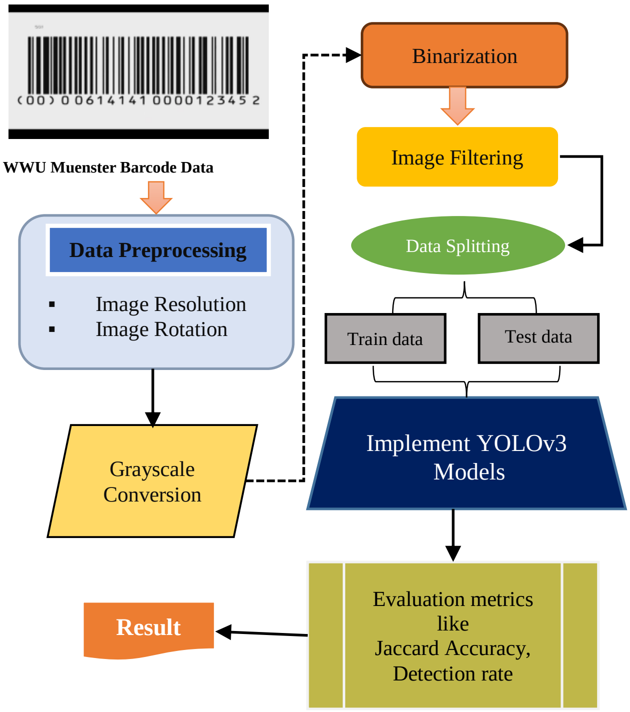

Fig. 1. Flowchart of 1D Barcode Scanning

The overall steps and process of methodology are briefly elaborated below:

A. Data Collection

The dataset is Wachenfeld's WWU Muenster Barcode Database, which will be shortened to "Muenster" from here on. A Nokia N95 phone was used to collect a total of 1055 EAN and UPC-A barcodes. For 595 pictures, it has the ground truth. The dataset picture sizes used in this experiment are 640 × 480. The histogram is provided below:

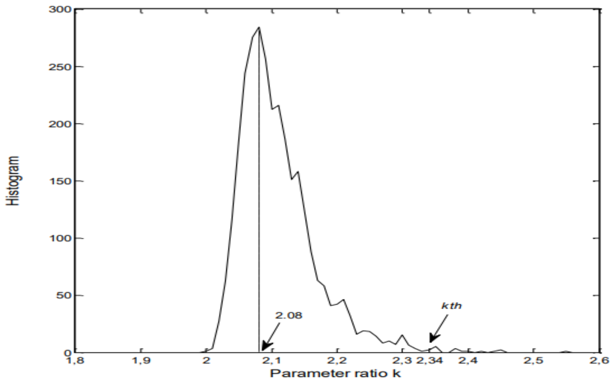

Fig. 2. Plot of Histogram of Parameter Ratio

Figure 2 presents a histogram of the parameter ratio k, illustrating the distribution of values with a smooth color gradient from blue to yellow. The plot highlights a prominent peak around k = 2.08, indicating the most frequently occurring value in the dataset. Additionally, an annotation marks a specific point labeled kth, suggesting its significance in the analysis. The smooth curve overlaying the histogram suggests an estimated density function, providing a clearer visualization of data trends. The use of a color gradient enhances the readability and visual appeal of the distribution.

B. Data Preprocessing

Preprocessing is a basic component of most automated image analysis systems. Basically, image preprocessing enhances image quality, reduces noise, and removes unwanted, superfluous artifacts from images. It helps improve the results of subsequent processes. The following steps og pre-processing are listed below:

- **Image Resolution:** All barcode images were resized to 640 × 480 pixels.
- **Image Rotation:** The mask picture rotated at the same angle as the 1D barcode color image θ. The rotated 1D barcode color picture served as the training data for this experiment.

C. Grayscale Conversion

This process transforms an image into different shades of gray, where each pixel represents intensity levels between black (0) and white (255). By eliminating color variations, grayscale conversion enhances edge detection, contrast adjustments, and feature extraction, which are essential for accurately detecting barcode regions in an image.

D. Binarization

Binarization is a technique that improves the contrast between barcode elements and their surrounds by transforming greyscale images into binary images, where pixels are categorized as either black (foreground) or white (background). Common binarization techniques include global thresholding (such as Otsu's method) and adaptive thresholding, which adjust pixel intensity values dynamically based on local regions. Effective binarization reduces noise and improves the accuracy and ensuring clear separation between bars and spaces in the barcode pattern.

E. Image Filtering

The purpose of applying a filter to an image is to enhance its quality. The correct detection of noise, correction of that noise, and smoothing of the picture must precede the filtering of the image [10]. The pixel quality of a picture can be improved with the use of image filtering. It includes the steps used to change pictures by adjusting their pixel values.

F. Data Splitting

The purpose of data splitting is to assess the efficacy of a model by making a training set and a testing set out of it. The model is trained using the test set, which is used to assess the model's generalisability and accuracy. The usual distribution is as follows: 80% for training and 20% for testing.

G. YOLOv-3 Model Implementation

The following generation, YOLOv3 [11], has substantial enhancements over YOLOv2 [12]. YOLOv2's eleven more object detection layers and Darknet-19 form its backbone. In contrast to YOLOv2, which struggles to recognise small objects, Figure 3 demonstrates how YOLOv3 uses residual blocks, skip connections, and upsampling to achieve state-of-the-art performance. Its backbone, Darknet-53, is said to be more effective than Darknet-19, ResNet-101, and ResNet-152 [13].

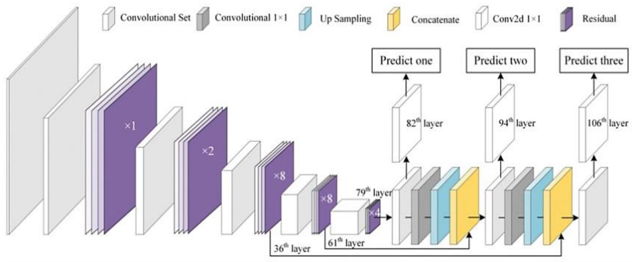

Fig. 3. YOLOv3 Neural Network Architecture

The ability to forecast three distinct scale levels is the primary differentiator between YOLOv3 and its predecessors. There is a 32-, 16-, and 8-fold down sampling of the input pictures, correspondingly. The initial detection occurs at the 82nd layer following the down sampling of the initial input from the preceding 81 levels. If the input picture is 416 × 416, the final feature map would be 13 × 13 × depth since the stride of the 81st layer is 32. After that, depth concatenation is used to connect the feature map from the 61st layer with the feature map from the 79th layer, which has been upsampled to dimensions of 26 × 26 × depth [14][15]. The second detection happens in the 94th layer, after Several convolutional layers have been applied to the composite feature maps. The identical process is carried out once more. Equation (1) may be used to determine the loss function L in YOLOv3:

$$L = Error_{boxes} - Error_{confidence} - Error_{classes} \quad (1)$$

where the bounding box coordinate regression is denoted by $Error_{boxes}$ which may be described as follows Equation (2):

$$Error_{boxes} = \lambda_{coordinate} \sum_{i=0}^{S^2} \sum_{j=0}^{N} l_{ij}^{obj} [(t_x - t_x')^2 + (t_y - t_y')^2 + (t_w - t_w')^2 + (t_h - t_h')^2 \quad (2)$$

where $S2$=S×S cells, the projected cells have N bounding boxes with the correct coordinates (tx, ty, th, tw), and $\lambda coordinate$ is the coordinate error weight. To determine the decline in trust in the bounding box's current item, the $Errorconfidence$ is utilized, and it may be described as follows Equation (3):

$$Error_{confidence} = \sum_{i=0}^{S^2} \sum_{j=0}^{N} l_{ij}^{obj} [c_i' \log(c_i) + (1 - c_i') \log(1 - c_i)] + \lambda_{confidence} \sum_{i=0}^{S^2} \sum_{j=0}^{N} l_{ij}^{confidence} [c_i' \log(c_i) + (1 - c_i') \log(1 - c_i) \quad (3)$$

where the confidence error weight is $\lambda conuidence$ and c is the number of classes..

H. Evaluation Metrics

Performance metrics are quantitative measures used to evaluate a model's effectiveness and accuracy. It help assess how well a model performs tasks such as classification, detection, and prediction. The following matrix that utilized the determine the performance of models are provided in below:

1) Jaccard Accuracy (J(avg))

Jaccard Accuracy, also known as the Jaccard Index, evaluates how well the expected and actual numbers match. It is used to evaluate object detection and segmentation models as shown in the Equation (4).

$$J(avg) = \frac{\sum_{i=1}^{N} \frac{TP_i}{TP_i + FP_i + FN_i}}{N} \quad (4)$$

Where:

- TP_i = True Positives (correct detections) for the ith sample
- FP_i = False Positives (incorrect detections) for the ith sample
- FN_i = False Negatives (missed detections) for the ith sample
- N = Total number of samples

2) Detection Rate

Detection Rate evaluates the percentage of barcodes in the dataset that were properly identified out of all the barcodes. Equation (5) provides the information:

$$D_{0.5} = \frac{\#(i \in S) | J(R_i, G_i) \geq 0.5}{|S|} \quad (5)$$

using S as the collection of dataset files. It present the detection for several accuracy levels in increments of 0.1 for completeness.

IV. Result Analysis And Discussion

This section explores a machine learning-based approach for detecting and recognizing 1D barcodes. The Python programming language is used to implement the experiment, which is operating on a PC with an Intel(R) Core (TM) 2 Duo CPU E7300 at 2.66GHz. Visual analysis through graphs and charts illustrates the effectiveness of the model in handling variations in illumination, distortions, and barcode positioning. Table II provides the proposed model performance.

TABLE II. Evaluation of Proposed YOLO Model for 1D Barcode Scanning on Muenster Barcode Database

Evaluation Metrics	YOLOv-3
Accuracy (Javg)	93.9
Detection Rate (D0.5)	99.8

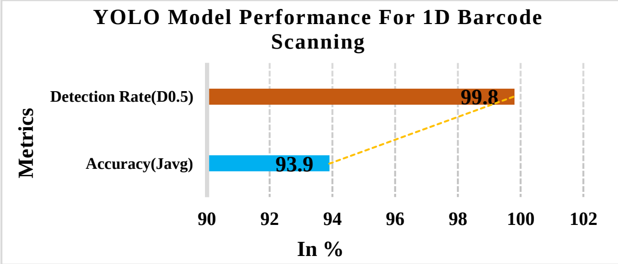

Fig. 4. Bar Graph for Performance of YOLO

Table II and Figure 4 present a horizontal bar chart displaying the evaluation metrics of a YOLO model used for 1D barcode scanning via machine learning. The Detection Rate (D0.5) is shown as 99.8%, indicated by a long, brown bar extending nearly to the 100% mark on the horizontal axis. The Accuracy (Javg) is represented by a shorter, light blue bar reaching 93.9%.

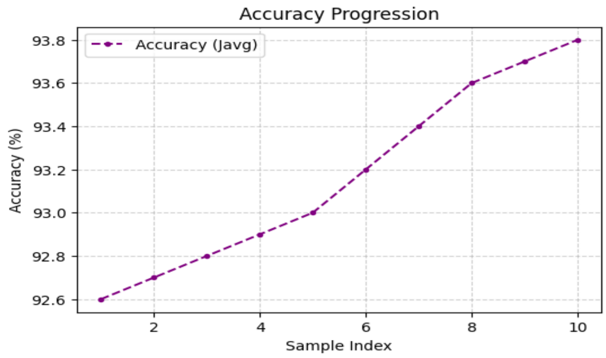

Fig. 5. Jaccard Accuracy graph for YOLOv-3 Model

Figure 5 depicts the accuracy progression of a YOLO model across ten sample indices. The y-axis shows the percentage of accuracy, which varies between 92.6% and 93.8%, while the x-axis shows the sample index. The purple line, labeled "Accuracy (Javg)," demonstrates a consistent upward trend, indicating improved model performance with each subsequent sample. Notably, the accuracy increases steadily, showcasing the capacity of the model to learn and improve as it consumes additional data. The presented graph shows successful YOLO model operation through its ongoing positive performance trend.

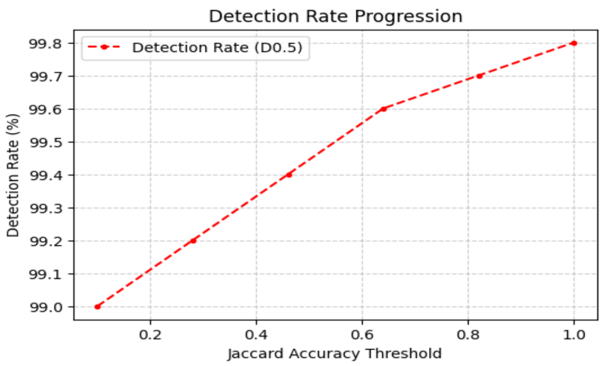

Fig. 6. Detection Rate Graph for YOLOv-3 Model

The detection rate of YOLO models shown in Figure 6 steadily increases according to the Jaccard Accuracy Threshold which ranges from 0.0 to 1.0. The detection rate (D0.5) in the graph increases from 99s.0% to 99.8% as measured on the y-axis. The Jaccard Accuracy Threshold appears on the x-axis as it measures the extent of prediction-ground truth box overlap. As the Jaccard Accuracy Threshold rises, the red line shows a continuous increase in detection rates, which signifies higher precision values. The detection precision of the YOLO model increases when the requirement for detection overlap becomes more rigorous.

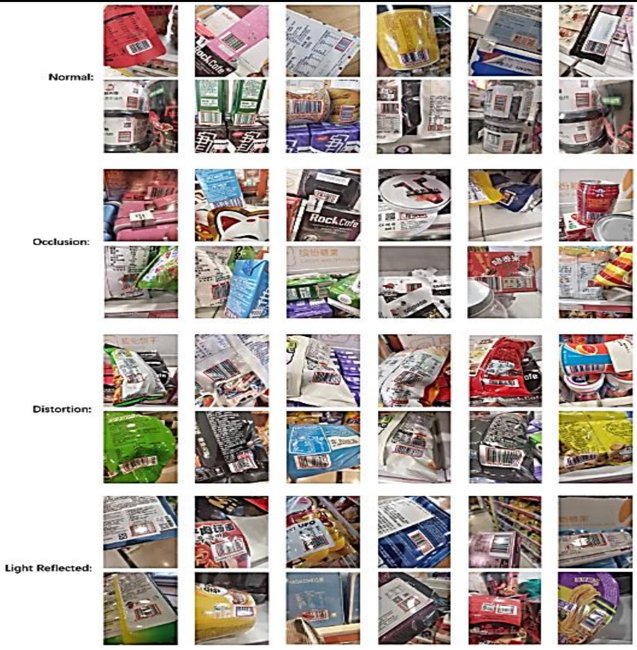

Fig. 7. Barcode Scanning Performance Under Different Conditions

Figure 7 displays barcode scanning results which group data points into four sections: Normal, Occlusion, Distortion, and Light Reflected. The Normal category represents barcodes that are visible without interference. Barcodes found in the Occlusion category become challenging to scan because objects or labels obstruct their visibility. Barcodes placed on curved or crumpled surfaces fall into the Distortion category which causes misalignment during scanning. Lastly, the Light Reflected category shows barcodes affected by glare, reducing readability.

A. *Comparative Analysis*

This section conducted a comparison of the metrics, accuracy, and detection rate with prior research that made use of the Barcode Database at WWU Muenster. Table III displays the outcomes of the comparison. The demonstration shows YOLOv3 stands as the most dependable solution for detecting and identifying barcodes because it achieves a 93.9% accuracy and a 99.8% detection rate. The Least Square Method achieves 88.2% accuracy and 96.6% detection rate which establishes itself as a strong performer yet YOLOv3 demonstrates superior capabilities. The strong detection output of YOLOv2 reaches 87.3% accuracy and 99.1% detection rate while its precision falls behind YOLOv3. Lastly, MSER records the lowest accuracy of 79.9% and a detection rate of 96.3%, making it the least effective among the models. While all models exhibit high detection capabilities, YOLOv3 proves to be the most efficient, making it the best choice for accurate and reliable barcode scanning.

TABLE III. COMPARATIVE ANALYSIS OF MACHINE LEARNING MODELS FOR BARCODE SCANNING

Model	Accuracy	Detection Rate
YOLOv3	93.9	99.8
Least square [16]	88.2	96.6
YOLOv2[17]	87.3	99.1
MSER[18]	79.9	96.3

The proposed approach offers several advantages for 1D barcode detection. YOLOv3 implements its multi-scale detection system along with residual blocks to establish unprecedented accuracy (93.9%) and detection rate (99.8%) that surpasses conventional techniques. The preprocessing pipeline provides robust barcode detection by enhancing visibility under all kinds of environmental conditions. YOLOv3's real-time processing ensures fast, efficient recognition, making it ideal for industrial and commercial applications like inventory management, retail, and logistics.

V. Conclusion And Future Scope

Barcodes are essential in retail, logistics, and inventory management, facilitating fast and accurate data retrieval. Traditional barcode scanning methods depend on specialized hardware, but ML-based detection and recognition have significantly enhanced their efficiency. The WWU Muenster Barcode Database was utilized in this paper, and comprehensive Preprocessing methods were used to improve barcode detection, including greyscale conversion, binarization, noise reduction, and picture filtering. The YOLOv3 model, leveraging multi-scale detection and residual blocks, achieved 93.9% Jaccard accuracy (Javg) and a 99.8% detection rate (D0.5), outperforming existing methods such as YOLOv2, Least Square, and MSER. Additionally, the model exhibited strong resilience to occlusions, distortions, and lighting variations, proving its effectiveness in real-world applications. While effective, the proposed method has limitations. YOLOv3's high computational cost makes it less suitable for edge devices. The dataset, captured under controlled conditions, may limit generalization to real-world scenarios. Additionally, barcode detection for rotated and skewed images can be improved.

Future work will focus on lightweight model optimizations for mobile deployment, expanding datasets for diverse conditions, and enhancing post-processing techniques like edge refinement and adaptive thresholding to improve accuracy.

References

[1] R. Brylka, U. Schwanecke, and B. Bierwirth, "Camera Based Barcode Localization and Decoding in Real-World Applications," in *2020 International Conference on Omni-Layer Intelligent Systems, COINS 2020*, 2020. doi: 10.1109/COINS49042.2020.9191416.

[2] Y. Xiao and Z. Ming, "1D Barcode Detection via Integrated Deep-Learning and Geometric Approach," *Appl. Sci.*, vol. 9, no. 16, Aug. 2019, doi: 10.3390/app9163268.

[3] V. Kulyukin and T. Zaman, "An Algorithm for in-place vision-based skewed 1D barcode scanning in the cloud," in *Proc. 18th Int. Conf. Image Processing and Pattern Recognition*, 2014, pp. 21–24.

[4] A. Sharif, J. Jia, J. Zhang, and G. Zhai, "Joint Barcode and Text Orientation Detection Model for Unmanned Retail System," in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, IEEE, Oct. 2020, pp. 1–5. doi: 10.1109/ISCAS45731.2020.9180932.

[5] I. Kalinov *et al.*, "WareVision: CNN Barcode Detection-Based UAV Trajectory Optimization for Autonomous Warehouse Stocktaking," *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 6647–6653, Oct. 2020, doi: 10.1109/LRA.2020.3010733.

[6] A. Burta, R. Szabo, and A. Gontean, "Object Recognition Development for Android Mobile Devices with Text-to-Speech Function Created for Visually Impaired People," in *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*, IEEE, Jul. 2020, pp. 608–613. doi: 10.1109/WorldS450073.2020.9210381.

[7] J. Jia *et al.*, "EMBDN: An Efficient Multiclass Barcode Detection Network for Complicated Environments," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 9919–9933, Dec. 2019, doi: 10.1109/JIOT.2019.2933254.

[8] H. Zhang, G. Shi, L. Liu, M. Zhao, and Z. Liang, "Detection and identification method of medical label barcode based on deep learning," in *2018 8th International Conference on Image Processing Theory, Tools and Applications, IPTA 2018 - Proceedings*, 2019. doi: 10.1109/IPTA.2018.8608144.

[9] N. N. Ventsov and L. A. Podkolzina, "Localization of Barcodes Using Artificial Neural Network," in *Proceedings of 2018 IEEE East-West Design and Test Symposium, EWDTS 2018*, 2018. doi: 10.1109/EWDTS.2018.8524837.

[10] D. Bhishman, U. Kushwaha, and S. Jha, "Image Filtering - Techniques, Algorithm And Applications," *GIS Sci. J.*, vol. 7, no. 11, pp. 970–975, 2020.

[11] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," Apr. 2018.

[12] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jul. 2017, pp. 6517–6525. doi: 10.1109/CVPR.2017.690.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[14] G. Krishna and J. Qaddour, "Implementation Problems Facing Network Function Virtualization and Solutions," in *ICN 2018 : The Seventeenth International Conference on Networks*, IARIA, 2018, pp. 70–76.

[15] R. Tarafdar and Y. Han, "Finding Majority for Integer Elements," *J. Comput. Sci. Coll.*, vol. 33, no. 5, pp. 187–191, 2018.

[16] A. Namane and M. Arezki, "Fast Real Time 1D Barcode Detection From Webcam Images Using the Bars Detection Method," *Lect. Notes Eng. Comput. Sci.*, vol. 2229, no. July, pp. 501–507, 2017.

[17] D. K. Hansen, K. Nasrollahi, C. B. Rasmusen, and T. B. Moeslund, "Real-Time Barcode Detection and Classification using Deep Learning," *Int. Jt. Conf. Comput. Intell.*, vol. 1, pp. 321–327, 2017, doi: 10.5220/0006508203210327.

[18] C. Creusot and A. Munawar, "Real-Time Barcode Detection in the Wild," in *2015 IEEE Winter Conference on Applications of Computer Vision*, IEEE, Jan. 2015, pp. 239–245. doi: 10.1109/WACV.2015.39.