IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 30 January 2024, accepted 9 May 2024, date of publication 14 May 2024, date of current version 21 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3400844

RESEARCH ARTICLE

Batch Normalization Free Rigorous Feature Flow Neural Network for Grocery Product Recognition

PRABU SELVAM[1], (Member, IEEE), MUHAMMAD FAHEEM[2], (Member, IEEE), VIDYABHARATHI DAKSHINAMURTHI[3], AKSHAJ NEVGI[4], R. BHUVANESWARI[5], K. DEEPAK[5], AND JOSEPH ABRAHAM SUNDAR[6]

[1]School of Computer Science and Engineering, VIT University, Chennai 600127, India
[2]School of Technology and Innovations, University of Vaasa, 65200 Vaasa, Finland
[3]Department of Computer Science and Engineering, Sona College of Technology, Salem 636005, India
[4]School of Computer Science and Engineering, VIT University, Vellore 632014, India
[5]Department of Computer Science and Engineering, Amrita School of Computing, Amrita Vishwa Vidyapeetham, Chennai 642109, India
[6]School of Computing, SASTRA University, Thanjavur 613401, India

Corresponding author: Muhammad Faheem (muhammad.faheem@uwasa.fi)

The work of Muhammad Faheem was supported by the Academy of Finland under Project WP3-Profi6(2708102611).

ABSTRACT Automatic product recognition is crucial in advancing economic and social fronts due to its superior reliability and time-saving nature compared to manual operations. The precise organization of products on store shelves is essential for boosting sales and ensuring customer satisfaction. However, verifying that the physical arrangement aligns with the ideal plan is a costly and time-consuming task for store personnel. In the computer vision domain, detecting products in scene images poses a considerable challenge, particularly when dealing with grocery items displayed on store shelves. The arrangement of products often presents crowded environments with numerous identical objects placed closely together. This study illustrates the ongoing challenge of identifying specific objects in complex situations despite using advanced object detection systems. The proposed framework consists of a three-stage pipeline. The initial stage incorporates a cutting-edge product detection algorithm, YOLOv5, to locate multiple grocery objects. The proposed OD-Refiner layer in the second stage identifies the missed retail object and rectifies overlapping bounding boxes of YOLOv5. The OCR-based object recognizer called Batch Normalization Free Rigorous Feature Flow Neural Network (BNFRNN) is proposed in the final stage of the pipeline. The performance of the proposed framework was evaluated using a benchmark dataset, WebMarket. The proposed framework outperforms current state-of-the-art approaches by achieving a precision score of 92.56%, recall of 85.64%, and F-score of 88.97%.

INDEX TERMS Convolutional neural network, deep learning, object detection, object recognition, text recognition.

I. INTRODUCTION

The retail industry faces a critical imperative—to augment the shopping experience for customers while optimizing business operations. In-store retail has evolved significantly, aligning with prior technological advancements such as retail product identification. Notably, barcode recognition stands out as the prevailing technology in use today. Its implementation has streamlined product administration and facilitated the advent of self-checkout systems. However, the placement flexibility of barcodes on products has inadvertently elongated the purchasing process, resulting in suboptimal retail experiences. Besides, the persistent challenge of supermarkets grappling with a substantial workforce to manage inventories and items remains unaddressed. Recent strides in Artificial Intelligence (AI) and Machine Learning (ML) present viable solutions

The associate editor coordinating the review of this manuscript and approving it for publication was Jenny Mahoney.

FIGURE 1. Challenges of on-shelf retail product recognition.

to these issues, promising further enhancement within the retail industry. Numerous merchants are allocating resources to AI initiatives, particularly in Product Recognition (PR). Businesses increasingly leverage AI technologies to overhaul the retail landscape, amalgamating online and offline operations [1].

According to a Juniper Research survey, merchants are projected to amplify spending on AI services in 2026 to more than threefold of the 2019 expenditure, escalating from $3.6 billion [2]. This forecast indicates that future retail operations may predominantly rely on ML and AI technologies. Due to an elevated standard of living, the burgeoning availability of a diverse range of retail items has necessitated a substantial workload and extensive human labor for product identification and goods management. In addition, the proliferation of innovative image-capturing devices has led to an exponential surge in digital content in the form of product images. Consequently, the accurate analysis and processing of vast visual data and the identification and classification of supermarket products have emerged as pivotal research challenges within the PR domain. ''Product Recognition'' denotes using computer vision technologies primarily grounded in computer vision to supplant manual product identification and classification. On-shelf product recognition systems can potentially address the challenges, as shown in Fig. 1.

Constructing an efficient PR system requires integrating two distinct yet interrelated technologies: object detection and recognition. This research delineates the pivotal steps involved in this process. Firstly, the foundation lies in employing a robust algorithm for product detection. Considerations for its implementation involve addressing potential inaccuracies stemming from densely packed items on shelves. This task is accomplished through various post-processing phases to rectify any erroneous detections. Furthermore, achieving real-time detection, vital for specific applications, demands the judicious selection of Deep Learning (DL) models tailored to meet these requirements. It assesses available competitive object detection frameworks that aid in optimizing performance within the proposed pipeline. Subsequently, object recognition, crucial for assigning class labels, forms the subsequent phase in the pipeline.

The SIFT and SURF algorithms have historically addressed this issue before the advancements in DL within computer vision. While this method can yield satisfactory outcomes, its resilience in real-world scenarios is limited, particularly with images captured in low-light conditions. Furthermore, the reliance on hand-crafted characteristics in these algorithms hinders their ability to capture pertinent data regarding product features. Alternatively, an ML-based Optical Character Recognition (OCR) System could be a viable option for object recognition. This approach involves assimilating and extracting all textual information on a product's packaging. The acquired text can then be cross-referenced with a database for recognition purposes. Notably, discrepancies in word recognition, such as misspellings, should not impede the matching process. However, when it pertains to the identification of retail products, challenges persist in image classification and object detection [3], [4], [5], [6], [7].

Over the past decade, DL has emerged as a predominant force in computer vision applications, particularly in classifying images and detecting objects. Unlike conventional methods reliant on manually crafted features, DL autonomously extracts features directly from image data [8], [9], [10]. This approach's efficacy lies in DL's capability to unearth intricate details through its deeper network layers. DL techniques exploit these advantages and offer innovative solutions to pivotal computer vision challenges such as picture segmentation and key point recognition. Recent endeavors in the retail sector [11], [12], [13] have demonstrated the promising outcomes achieved by employing DL-based detectors, substantially enhancing efficiency and accuracy in identifying real-world objects. However, despite remarkable strides in object identification technology, the complexity persists when detecting items in densely populated scenes, posing challenges even for state-of-the-art object detectors. This challenge propelled our focus toward developing algorithms to detect and recognize similar-looking or identical items positioned closely together—an aspect often overlooked in prevailing object detection frameworks.

Consequently, these sophisticated object detectors tend to falter when faced with such scenarios. For instance, discerning a precise item becomes arduous when two related products share shelf space, leading to challenges in minimizing the overlap between the Bounding Boxes (BB) and delineating these items' boundaries. Advanced models like RetinaNet [14] frequently encounter difficulties, generating BBs that inadequately capture multiple items or erroneously identify adjacent item regions as distinct entities.

This paper delves into the exploration of OCR-based technology in object recognition, addressing the limitations of conventional OCR algorithms when confronted with intricate tasks such as object identification. While traditional OCR methods exhibit proficiency in handling articles, their accuracy substantially diminishes in more complex scenarios. The challenges stem from diverse text sizes, styles, layouts, and

unfamiliar imaging conditions, often introducing aberrations like specular reflection, shadows, occlusion, and motion blur into the final image. To surmount these obstacles, the integration of visual saliency becomes imperative to pinpoint text segments within an image. Leveraging deep Convolutional Neural Networks (CNNs) has showcased considerable success in text recognition within images. Nonetheless, prevalent CNN-based models possess inherent drawbacks. Classical convolution suffers from shortcomings in effectively utilizing hierarchical features across multiple layers during optimization, leading to the loss of crucial feature information and hindering learning within the algorithm.

In response to these limitations, this research introduces a novel three-stage pipeline model tailored for recognizing on-the-shelf and off-the-shelf grocery objects:

- The proposed framework employed the state-of-the-art object detection algorithm, ''YOLOv5,'' to identify multiple and small retail objects.
- This paper introduces the ''OD-Refiner'' stage, comprising two refinement procedures: redundant box removal and missing box removal, aimed at circumventing the deficiencies of the object detection model.
- The final phase of this study's pipeline focuses on the object recognition phase. A new text recognition model, ''Batch Normalization Free Rigorous Feature Flow Neural Network,'' is proposed to capture complex shape text information such as product name, brand name, expiry date, and so on from the retail objects package.
- The efficacy of the proposed framework is evaluated based on metrics including precision, recall, and F-score. Comparative analysis with other leading models is given in the graph, and the table demonstrates the proposed model's superiority, maintaining a substantial advantage over its contemporaries.

The paper's structure is as follows: Section II introduces the literature survey, and the proposed three-stage pipeline in Section III. Section IV delineates the performance analysis, and finally, Section V serves as the conclusion and future work of the paper.

II. RELATED WORKS

Theoretically, the challenge of automatically identifying grocery items from photographs originates from visual object identification, a more extensively studied area [15]. It was observed that handling supermarket items on shelves involves peculiarities that significantly increase the complexity of this task. Extensive efforts have been dedicated to enhancing the automated visual recognition of grocery items [16], [17], [18].

The Shelf-Scanner, a groundbreaking model in shelf recovery developed by Winlock et al. [16], operates as a real-time video feed PR system. Recommendations for the Shelf-Scanner involve visually representing the shelf's contents by scanning objects. To cater to visually impaired individuals, López-de-Ipiña et al. [19] introduce a shopping system utilizing technologies such as RFID and QR codes, which operate through supporting gadgets. MyHalal, proposed by Kassim et al. [20], utilizes smartphone camera equipment and barcode scanners to determine a product's Halal status. Despite advancements in barcode scanners, the research mentioned requires human operator intervention. Automatic PR often encounters challenges such as few-shot or single-matching scenarios due to the difficulty in gathering necessary training data. Previously, hand-crafted feature approaches like Histogram of Oriented Gradients (HOG) [21] and Speeded Up Robust Features (SURF) [22] held a strong reputation. However, DL methods have progressively surpassed them by enhancing image identification and object detection accuracy.

A cutting-edge object detection technique was proposed by Marder et al. [23] to scan and assess retail products exhibited in grocery stores through image analysis. This method utilized a generic product detection module [24] in conjunction with support vector machines [25] to tackle the issue of product recognition. This approach effectively breaks down PR into two distinct components: object detection and recognition. George et al. [26] initially highlighted that brand logos typically encompass textual elements, forming a primary characteristic of objects. Using text recognition, they employed active learning-based classification to categorize products at the brand level. However, their method cannot classify products at the individual product level, and text recognition proves ineffective when a brand lacks textual elements.

An alternate approach by Tonioni et al. [27] viewed identifying items in a grocery store as a subgraph isomorphism problem, incorporating a planogram. While this method yields remarkable outcomes, its implementation is challenging due to the continual changes in the planogram. Similarly, Geng et al. [28] conducted product detection and recognition employing a coarse-to-fine perception strategy, utilizing feature-based matching and a singular deep learning technique. Leveraging DL's robust capabilities, their method distinguishes between similar products with minor differences. Most automated methods, including those mentioned above, treat product recognition as a single-stage investigation within the broader scope of multi-object detection. To effectively implement a practical identification system, addressing the frequent changes in the products offered for sale in a store and their display is essential.

Tonioni et al. [11] tackled this challenge by developing a DL pipeline that commences with initial product-agnostic object detection. Subsequently, they progress towards PR by comparing global descriptors computed from reference images with cropped query images, utilizing innovative object detectors. Using minimal training data, precisely one image per category, Karlinsky et al. [12] introduce a non-parametric probabilistic model to detect and recognize hundreds of fine-grained item categories within static images. The phase 1 detection and recognition pipeline performs better than existing methods on established benchmarks and

their respective datasets. This phase's efficacy is further augmented by employing a deep network that leverages finely tailored training data synthesized and integrated from phase 1 outputs. Moreover, implementing simple tracking techniques boosts performance when handling static images.

In the realm of categorizing structured objects within datasets containing numerous visually similar categories, Goldmana et al. [29] present a novel DL architecture employing CNNs. This architecture learns the parameters of a Conditional Random Field (CRF) by incorporating local visual features and neighboring classes, fostering collaborative learning for CRF parameter estimation. The model addresses the challenge of comprehensively understanding contextual relationships in scenarios with abundant classes and limited data, surpassing the limitations of conventional CRF approaches. Meanwhile, Qiao et al. [13] investigate the generation of object proposals from images captured in supermarkets and natural settings. The proposed supermarket datasets comprise two real-world datasets alongside one synthetic dataset. ScaleNet, a technology for predicting item scales, innovates an object proposal framework. Experimental results indicate that the model, developed solely using MS COCO and synthetic supermarket datasets, exhibits equal efficacy when applied to the two real-world datasets. The study proposes an intelligent, unstaffed retail shop plan based on AI and the Internet of Things to explore the feasibility of implementing this shopping style. It involves the development of an end-to-end classification model utilizing the MASK-RCNN method, which leverages 11,000 images across various scenarios, encompassing 10 distinct types of Stock Keeping Units (SKUs).

To handle domain shifts during training, a Generative Adversarial Network (GAN) is employed alongside a deep CNN trained on samples generated by the GAN, as suggested by Tonioni et al. [30]. This combined architecture embeds images of products, establishing a relationship order among detected product categories. During testing, recognition is performed using a K-Nearest Neighbor search against a database of just one reference image per product. Additionally, Biasio et al. [31] advocate for a novel approach to retail shelf analytics in a real-world business setting by employing cutting-edge DL and image processing techniques. The prediction of product BBs from an image classification dataset is explored through a weakly supervised method detailed in [32], which relies on two algorithms. The initial algorithm involves a straightforward, Fully Convolutional Network (FCN) trained explicitly for object instance categorization.

Subsequently, the second approach employs a Convolutional Encoder-Decoder (ConvAE) to refine the FCN output mask and generate the final output BBs. ConvAE is trained using a deliberately constructed dataset of output segmentation masks to locate objects effectively. In a related study by Yi et al. [33], emphasis is placed on employing CNN to recognize 324 distinct goods in situ without requiring additional labeling of image-bounding boxes. An in-vitro dataset generates BBs, followed by an occlusion simulation method to test this hypothesis. Moreover, the effectiveness and efficiency of these methodologies in enhancing the Faster RCNN detection model have been substantiated. It has been established that implementing these methods leads to the development of an improved Faster RCNN detection model, applicable not only in this specific context but potentially in various other domains.

Franco et al. [34] outlined a three-step approach. Firstly, they engaged in candidate pre-selection, segregating foreground and background through fixed-threshold binarization. A fine-selection phase ensued, utilizing a customized deep neural network and a bag of words to identify the most resilient features. Finally, post-processing techniques were employed to diminish false positives by eliminating multiple overlapped detections of the same products. Bukhari et al. [35] engineered an automated retail checkout system reliant on vision technology, employing CNN for object detection. The system implements the Canny edge detector and hysteresis thresholding to execute NMS and generate a binary image highlighting edges. Additionally, morphological operations are conducted to address voids and spaces within the image. It is important to note that this approach relies significantly on a conveyor belt mechanism powered by a motor.

Ciocca et al. [36] introduced a multi-task learning network to extract features from images and conducted product classification using supervised and unsupervised learning methods. Similarly, Yilmazer and Birant [37] amalgamated semi-supervised learning with on-shelf availability concepts to detect empty shelves. Santra et al. [38] utilized a graph convolutional network (GCN) and Siamese network to extract features and capture similarity among neighboring superpixels. Finally, the features obtained from the GCN and Siamese network were employed in an SSVM to address identity gaps on the rack. Leo et al. [39] evaluated various classification models. Olóndriz et al. [40] introduced the FooDI-ML dataset and the Glovo application, aiming to identify retail product information.

Machado et al. [41] also devised a system for recognizing products intended for visually impaired individuals. Their findings indicated that the ResNet-50-based approach outperformed other deep learning models. Furthermore, Selvam and Koilraj [42] developed a deep-learning framework to identify retail objects. The framework comprises of object detection, text detection, and text recognition models. Gothai et al. [43] delineated the product recognition issue into dual stages: product detection and subsequent recognition. Initially, the YOLOv5 algorithm underwent customization for object detection. This adaptation enabled the extraction of specific attributes, such as shape, colour, and size, crucial for object recognition. These attributes were used with bag-of-words, gaussian shift theorem, and Naïve Bayes techniques to facilitate object recognition.

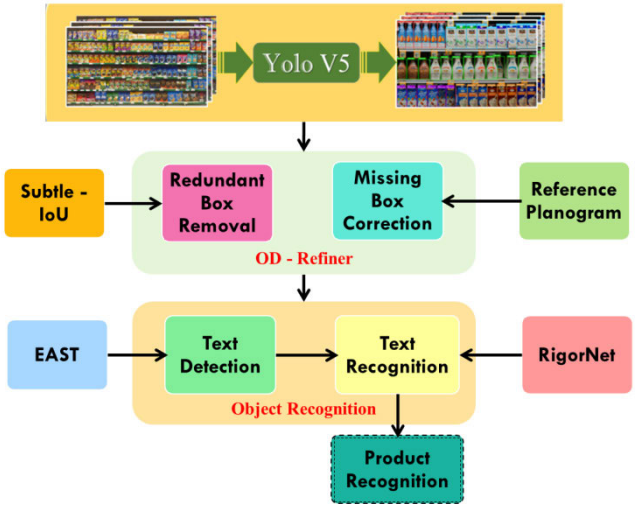


FIGURE 2. Pipeline architecture of the proposed retail object recognition system.

In reviewing the literature, several research gaps have been pinpointed. Current methodologies encounter challenges locating dark retail items, distinguishing between non-identical objects, experiencing increased partial detections, and encountering difficulties in finding multiple grocery items and objects within complex backgrounds.

III. PROPOSED FRAMEWORK

The architecture diagram of the proposed grocery object recognition framework is presented in Fig. 2. This system contains three stages: (i) object detection, (ii) OD-Refiner, and (iii) object recognition. Each of these stages is discussed in detail below.

A. GROCERY PRODUCT IDENTIFICATION USING YOLOv5

The YOLOv5 model integrates CSPDarknet and Path Aggregation Network (PANet), streamlining object detection training and reducing computational costs. Compared to other models, YOLOv5 excels in detecting smaller or distant objects. It exhibits favourable inference speed compared to Faster-RCNN, Fast-RCNN, and SSD, unlike R-CNN and SPP-net, overlapping boxes around retail objects.

Initially, CSPNet became part of the darknet architecture, forming CSPDarknet. This incorporation effectively tackles repeated gradient information issues commonly seen in extensive backbones. It incorporates gradient changes into feature maps, substantially enhancing CNN's learning capability. However, its lightweight nature causes a slight accuracy lag while significantly cutting unnecessary energy consumption by distributing computation across CNN layers. CSPDarknet reduces the model size by compressing feature maps through cross-channel pooling during feature pyramid generation, which is vital for our grocery product detection task.

Secondly, the YOLOv5 algorithm adopts PANet as its neck to augment information flow. PANet employs bottom-up path

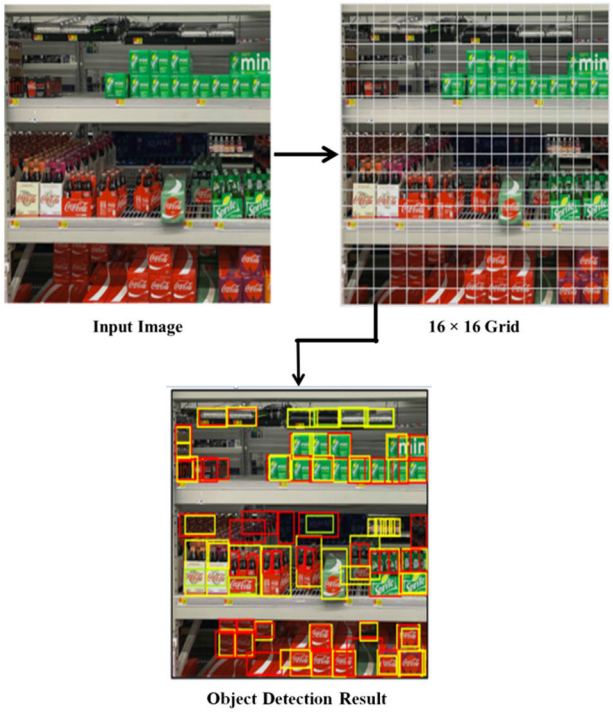


FIGURE 3. Sample result of the on-the-shelf retail objects using YOLOv5 algorithm.

augmentation and a new FPN, boosting hierarchical feature localization. Adaptive Feature Pooling enables high-level features to access fine details and high localization from low-level features. Simultaneously, large receptive fields capture richer context information in high-level features for precise predictions. A fully connected fusion aids mask prediction, distinguishing instances and identifying parts of the same object. PANet aids in identifying smaller grocery products by utilizing shared pooling features, ensuring product recognition without misses.

Finally, the YOLOv5 algorithm tailors the YOLO layer for multi-scale prediction, generating feature maps of varying sizes like 19 × 19, 38 × 38, and 76 × 76. This flexibility enables the model to handle and detect objects of different sizes – small, medium, and large, as shown in Fig. 3. It also predicts anchor boxes for feature maps. Given the diverse sizes of grocery products, the YOLO layer's multi-scale detection mechanism ensures consistent product detection, even if the size changes during the detection process.

The input image is divided into N grids, each with an $S \times S$ sized equal-dimensional region (see Fig. 4). There are N grids, each in charge of detecting and locating the object contained within it. As a result, these grids forecast the BB coordinates concerning their cell coordinates and the item label and likelihood that the object is present in the cell. In addition, each grid has a class probability. Then, the final output is a $S \times S \times (B * 5 + C)$ tensor. This method reduces computation by using cells from the image to handle both detection and recognition. Still, it generates many duplicate predictions since numerous cells predict the

same object yet have distinct BB predictions for each cell. Therefore, YOLOv5 uses Non-Maximal Suppression (NMS) to deal with this problem.

However, even though YOLOv5 is the best algorithm for detecting objects, it has significant drawbacks. Due to the grid constraint, YOLOv5 has difficulty detecting and classifying products appearing in shelf images. YOLO has difficulty detecting and locating too small objects that naturally occur in groupings, such as candy boxes. YOLO also has inferior accuracy compared to object detection algorithms like Fast RCNN, which operate at a reduced speed. Meanwhile, let's review some post-processing phases combined as the "Object Detection Refiner" (OD Refiner) following the object detection step to surpass the constraints of the YOLOv5 detection module before delving into specifics about the object recognition methodology.

B. OD-REFINER

The second stage of the proposed system is "OD-Refiner," which includes a set of processing steps, such as removing the redundant bounding boxes and fixing the missing box that runs simultaneously. This section presents the functionality of both techniques in detail.

1) REDUNDANT BOUNDING BOX REMOVAL

The YOLOv5 algorithm produces numerous irrelevant or redundant bounding boxes. Consequently, a method must be implemented to eliminate these irrelevant bounding boxes. Initially, the elimination process involves removing all irrelevant boxes unlikely to detect an object effectively. This redundancy reduction can be accomplished by establishing a Boolean mask ($tf.boolean_{mask} inTensorFlow$), retaining only those boxes surpassing a predefined probability threshold. This stage effectively eliminates any aberrant object detections. Despite this filtering, multiple boxes persist for each identified object. Nevertheless, the search for the appropriate box narrows down to one. YOLOv5 employs NMS to determine this bounding box.

2) NON-MAX SUPPRESSION (NMS)

Non-max suppression makes use of "Intersection over Union (IoU)." When given two boxes as input, it computes the intersection and union of the two boxes. Only a handful of approaches demonstrated substantial enhancement through adopting NMS (see Table 1), and specific methods proved computationally demanding. In grocery scenarios with densely packed objects causing numerous overlapping detections, the challenge of resolving detection ambiguities tends to increase rather than diminish.

The bounding box data is represented as $[class, x_{center}, y_{center}, width, height]$ in YOLOv5 object detection, alongside an objectness score (confidence) label, denoted as c within the range [0,1]. As elaborated further, during the training of the proposed Subtle-IoU layer, mitigating bias induced by noisy detections becomes essential. Thus, detections for which $c \leq 0.1$ are appropriately filtered.

TABLE 1. Algorithm of non-max suppression (NMS).

Algorithm 1: Non-Max Suppression (NMS)
outputs = []
bboxes = sort_bounding_boxes_by_score(bboxes_list)
while bboxes is not empty:
highest_score_box = get_highest_score_box(bboxes)
outputs.append(highest_score_box)
remaining_boxes = []
for box in bboxes:
iou = calculate_iou(highest_score_box, box)
if iou < iou_threshold:
remaining_boxes.append(box)
bboxes = remaining_boxes
return outputs

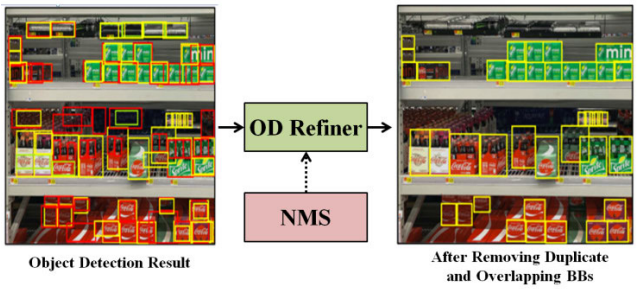


FIGURE 4. Sample result after removing redundant and overlapping bounding boxes.

3) SUBTLE-IOU LAYER

The NMS model is employed to compute objectness scores (O) in non-dense situations, which can resolve overlapping detections. Many products earn high objectness ratings in dense images because of the overlapping BBs. Each BB was recommended an extra value to deal with noisy positive detections [29]. To estimate the Subtle-IoU score by a fully convolutional layer, $O^{iou} \in [0, 1]$ was introduced as a third head at the tail of each region proposal network in the detector. To find the IoU between an estimated BB (b_j), where $j \in \{1..N$ and the actual BB, ($\hat{b}_j$), for the given N anticipated detections are provided. Then $\hat{b}_j$ is selected as the annotated box that is most closely related to b_i (in image coordinates). As long as they do not intersect, $IoU_j = 0$, the pixels are counted using $Intersection(\cdot)$ and $Union(\cdot)$.

$$IoU_j = \frac{Intersection\left(\hat{\boldsymbol{b}}_j, \boldsymbol{b}_j\right)}{Union\left(\hat{\boldsymbol{b}}_j, \boldsymbol{b}_j\right)} \tag{1}$$

The Subtle-IoU layer is used to learn a probabilistic interpretation of Eq. (1) using a binary cross-entropy loss.

$$\mathcal{L}_{sIoU} = -\frac{1}{n}\sum_{j=1}^{n}\left[IoU_j log\left(O_j^{iou}\right) + \left(1 - IoU_j\right) log\left(1 - O_j^{iou}\right)\right] \tag{2}$$

FIGURE 5. Sample result obtained after OD-Refiner operation.

The Subtle-IoU layer is used to learn a probabilistic interpretation of Eq. (2) using a binary cross-entropy loss that is represented as.

$$\mathcal{L} = \mathcal{L}_{Cl} + \mathcal{L}_{Reg} + \mathcal{L}_{sIoU} \tag{3}$$

In the Eq. (3), $\mathcal{L}_{Cl}$ is the standard Euclidean, class loss and $\mathcal{L}_{\text{Reg}}$ is the cross-entropy loss.

Fig. 5 shows the working mechanism of the Subtle-IoU layer. The procedure begins with taking the input image from the object detection layer. It subsequently transfers this image to the subtle layer alongside BB and objectness. This layer transforms the object detection module output into a Gaussian heat map representation. This representation identifies products detected as multiple and overlapping BBs. Ultimately, it assesses the clusters and produces a singular detection for each item.

4) MISSING BOX CORRECTION

The YOLOv5's object detection sometimes leads to missed product items and inaccurate detections. This issue primarily arises when identical products are closely arranged on shelves. Nonetheless, the initial phase can gather enough accurate detections to enable subsequent steps to identify the detected frame area. In turn, it allows restrictions on the expected product placements, significantly improving precision and recall.

In the first phase, a collection of bounding boxes is generated, corresponding to the detected instances of products. Within the second phase subsection, known as ''Missing box correction,'' the process commences by utilizing planogram information concerning items and their spatial arrangement. The approach chosen involves representing the planogram as a grid, functioning as a fully connected network. Each product face correlates with a node linked to a maximum of eight neighbouring nodes at a one-edge distance.

Opting for a graph instead of a grid allows for a more flexible representation of the planogram. It enables the depiction of the item's exact placement not through an edge connecting nodes but by signifying proximity within that path. This representation termed the reference planogram, transforms data concerning the count of faces associated with each product and the closeness of objects on shelves.

Table 2 demonstrates the proposed algorithm for computing sub-graph isomorphism between the reference and observed planograms. The initial phase's detections are utilized in this process, automatically constructing a grid-like graph mirroring the structure of the reference planogram, which becomes the observed planogram. The subsequent task involves establishing the isomorphism between these planograms to detect local clusters. This phase rectifies discrepancies in the observed planogram, often stemming from false detections in the initial step. Post this phase, the observed planogram exclusively retains genuine detections. The third phase, product verification, verifies whether items are misplaced or absent in the image. Consequently, it addresses a relatively more straightforward computer vision

TABLE 2. Algorithm of sub-graph isomorphism calculation.

Algorithm 2: Sub-graph isomorphism calculation between Reference Planogram and Observed Planogram
Input: Reference Planogram (I) and Observed Planogram (O)
Output: Output image with precise object detection
1. **Initialize** $C_{max} := 0$, $S_{best} := null$;
2. **Create** a preliminary hypothesis set. **Call** $BuildHypotheses\ (I, O)$ $H = \{\cdots h_i \cdots\}, h_i = \{n_I, n_O, c(n_I, n_O)\}$.
3. **Identify** S by repeatedly selecting the highest score hypothesis. **Call** $GetSolution(H, C_{max}, \tau)$;
4. **Eliminate** the hypotheses comprising whichever of the two nodes in the preeminent theory and escalate the scores of views connected with rational neighbours. **Update** H
5. Using the cardinality of S, along with penalizing factor for the occurrence of O F-score confidence score
6. Take the current best solution, C_{max} as input and to speed up the processing, calculate it using the Branch-and-Bound (BnB) scheme. **Impart** Branch-And-Bound (C_{max})
7. Upon returning from GetSolution, check whether or not the new solution S improves the best one found so far and remove h_0 from H and return C_{max}, S_{best}.
8. **Return** C_{max}, S_{best}

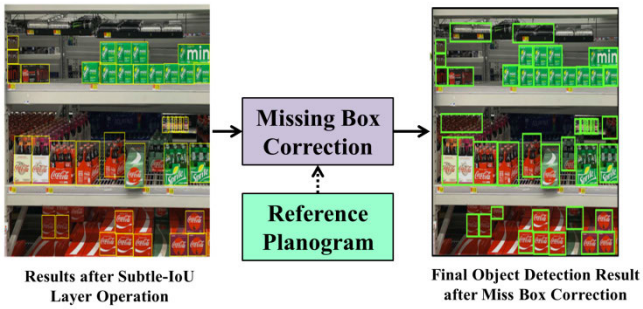


FIGURE 6. Final object detection result after missing box correction operation.

problem than the previous stage, demonstrating the presence or absence of a recognized object within a defined region of interest in the image (see Fig. 6).

Upon confirming the product's existence, the corresponding node joins the observed planogram, adding constraints among identified items. Otherwise, a planogram compliance issue linked to the observed node emerges. This iterative process continues until all detected shelf items align with discovered occurrences or are flagged as compliance violations. To elaborate, two planograms—the reference and observed—aim to identify an isomorphism between a subset I of the planograms, where subset O attains the largest possible cardinality while complying with product placement requisites in the latter. Only nodes in I and O, sharing the same product instance and coherent neighbours, are connected. It seeks to unveil the maximal number of self-consistent nodes in Graph O, aligning their relative positions with those in Graph I.

C. OBJECT RECOGNITION PHASE

The subsequent phase involves object recognition, constituting the model's second workflow stage. Employing a DL-based optical character recognition system could serve as an alternative for executing object recognition. This system would encompass all textual information present on the product's packaging and extract it. Initially, a text detection technique like Zhou et al.'s [44] Efficient Accurate Scene Text (EAST) detector gets utilized to identify the bounding box image. The manuscript can discern the object's text by cross-referencing it with a database through this text recognition model. Once the object's text is captured, it becomes storable in our text database. This paper introduces a novel deep network model termed Batch Normalization Free Fully-Convolutional Rigorous Feature Flow Neural Network (BNFRNN) designed for item detection based on text detected within the bounding box. The comparison of text with database content occurs through a process resilient to word recognition and spelling errors.

1) TEXT DETECTION USING THE EAST ALGORITHM

The text detection pipeline consists of two stages: eliminating unnecessary steps and intermediaries. Through the fully convolutional network, text prediction at the word/line level is feasible directly. The resulting predictions, which could be rotated rectangles or quadrangles, undergo processing in the NMR stage to derive the outcome (see Fig. 6). In this particular model, predictions of text instances and their configurations originate directly from the entirety of the images. The model, a neural network for text detection, adopts a fully convolutional structure to generate text regions as its output. Post-processing methods involve solely thresholding and NMS on expected geometric shapes.

2) TEXT RECOGNITION USING BATCH NORMALIZATION FREE RIGOROUS FEATURE FLOW NEURAL NETWORK (BNFRNN)

The information for text detection is subsequently utilized within the proposed recognition model named "BNFRNN" to recognize the detected text and corresponding objects precisely. BNFRNN represents an advancement over the current model, IntensiveNet [45]. This segment introduces the learning framework of BNFRNN. The complete learning architecture of BNFRNN is presented in this paper. Fig. 7 illustrates the comprehensive word recognition structure of BNFRNN for images of retail objects.

This study affirms that the proposed BNFRNN constitutes a fully convolutional framework comprising two rigorous layers and a manuscript layer (Taylor-SoftMax and CTC) at its core: Comp's initial layer, Layer1, extracts shallow features. Furthermore, Comp Layer1 operates as a downsampling function when dealing with more sizable elements. This particular layer receives input and extracts the subsequent properties.

$$F_0 = H_{Comp1}(input) \tag{4}$$

In this case (see Eq. (4)), $F_0 = H_{Comp1}(\cdot)$, which is a 3-layer composite consisting of a Rectified Linear Unit (ReLU) [46], followed by a 3×3 convolution layer and a learnable scalar multiplier (α)] [47]. All the separable composite functions in this study refer to a collection of operations consisting of ReLU, depth detachable convolution, and a single learnable scalar, as shown in Fig. 8. Recently, researchers have demonstrated that the order in which ReLU, convolution, and Batch Normalization (BN) are applied can improve the quality of the final product. However, BN has three key practical drawbacks. Due to the high cost of the computational primitive and the associated memory overhead [48], gradient evaluation can take much longer in some networks when using the weight normalization technique [49]. This model can train more complex neural networks by using skip connections and residual branches in combination with a normalization layer.

Due to batch norms, the scale of activations in the residual structure will be reduced. Due to this biasing effect, the network will have well-behaved gradients early in training, making optimization easier. The batch norm reduces the residual branch by $\sqrt{d}$ on average, where d is the total number of blocks in the residual branch. Instead of batch normalization, this work can use a learnable scalar multiplier α that

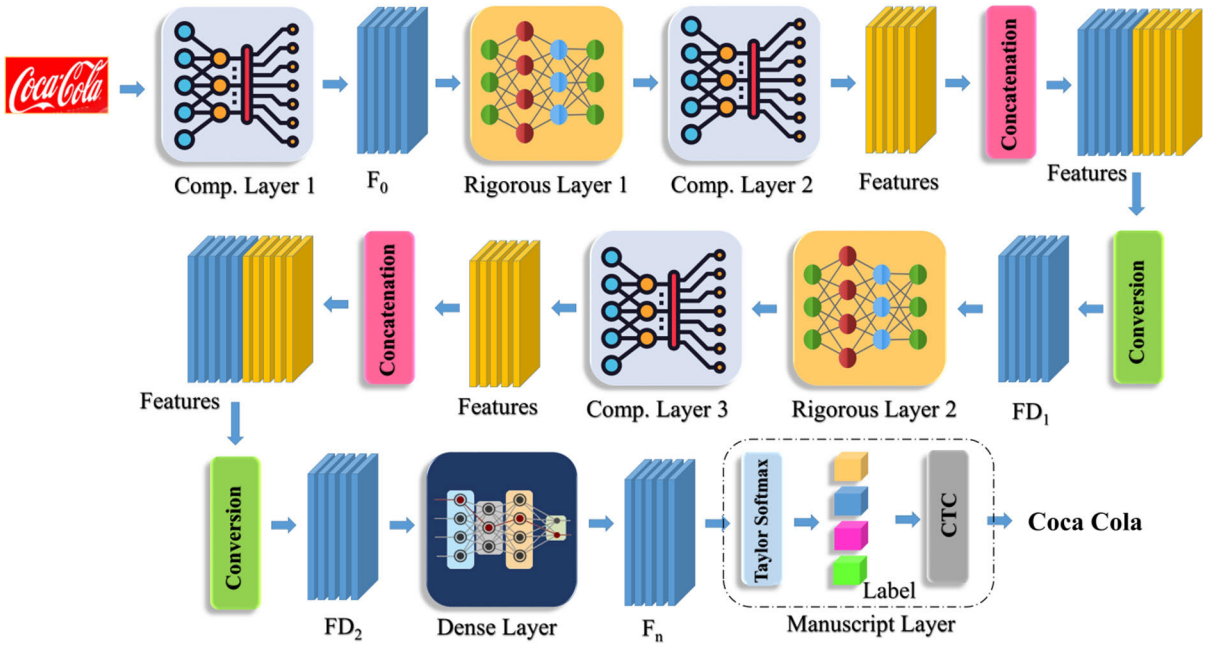


FIGURE 7. Pipeline architecture of the BNFRNN network.

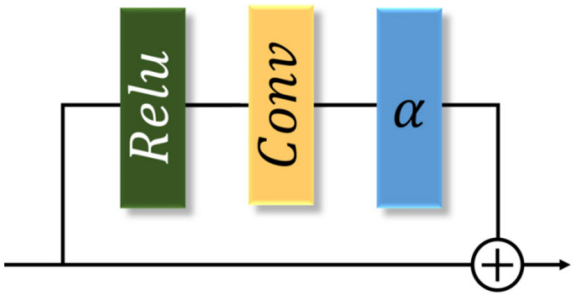


FIGURE 8. Composite function.

can be initialized to a small value, such as $\alpha <= 1/\sqrt{d}$. The unnormalized network can then be trained to go to great depths. This study can confirm that reducing the activation threshold by a certain amount aids the network's ability to learn quickly, even with numerous layers. Rigorous Layer1 takes the features from the previous layer and generates the high-mobility features FD_1.

$$FD_1 = H_{FDL1}(F_0) \tag{5}$$

In the Eq. (5), $H_{FDL1}(\cdot)$ represents the function of Rigorous Layer 1. The FD_1 includes the Rigorous Layer 1's final features that pass through the influence of the Dense layer present in the Rigorous Layer 1, which also contains several convolutions and concatenating operations. Correspondingly, FD_1 are then passed as input to the rigorous Layer 2. This work can obtain features FD_2 after the Rigorous Layer2 as (see Eq. (6)).

$$FD_1 = H_{FDL2}(FD_1) = H_{FDL2}(H_{FDL1}(F_0)) \tag{6}$$

In the above equation, $H_{FDL1}(\cdot)$ is the Rigorous Layer 2's function. When matched to the conventional dense layer, the extracted features are more dynamic and thus better suited for feature fusion. Finally, this work adds a dense layer to the proposed network to increase the network's ability to represent features. F_n is the final feature map obtained using the dense layer shown in Eq. (7).

$$F_n = H_{dense5}(FD_2) \tag{7}$$

In which $H_{Dense5}(\cdot)$ is the function of the final dense layer. The Taylor-SoftMax classifier [50] is then fed with the F_n Features. As a result, this work may derive the expected label as given Eq. (8).

$$label = Taylorsoft(F_n) = \sigma(z)_j = e^{z_j} / \sum_{k=1}^{K} e^{z_k} \tag{8}$$

where $\sigma(\cdot)$ is the Taylor-SoftMax function. In the OCR problem, this model uses CTC to turn the classifier's predictions into the final label sequence. The proposed rigorous layer structure is given in the Fig. 9.

$$\begin{aligned} Fd_2 &= H_{\text{dense2}}(Fd_1) \\ &= H_{\text{concat}}(Fd_1, Fd_{11}, Fd_{12}\ldots Fd_{1c}) \\ &= [Fd_1, H_{\text{conv}}(Fd_1), H_{\text{conv}}(Fd_1 + H_{\text{conv}}(Fd_1)), \\ &\quad \ldots H_{conv}(Fd_1 + H_{conv}(Fd_1) + \ldots + H_{conv}(Fd_{1c-1}))] \end{aligned} \tag{9}$$

3) RIGOROUS LAYER

The proposed rigorous layer includes three components: an integrated dense layer, a dense integration operation, and a manuscript layer.

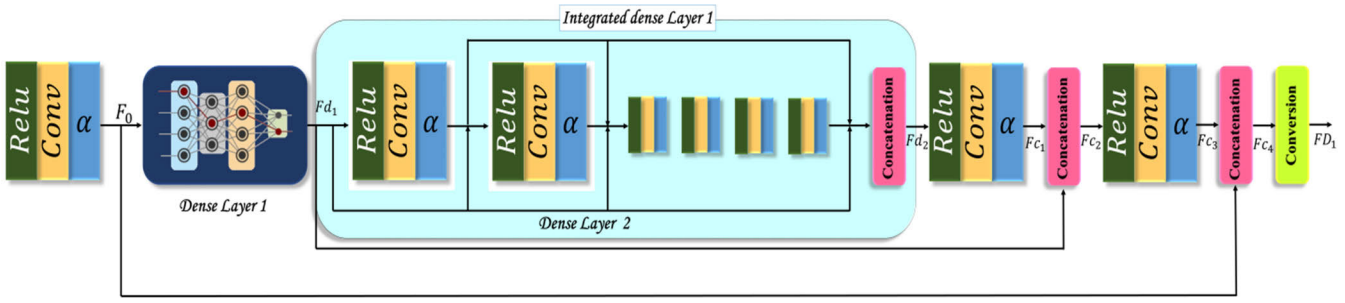


FIGURE 9. Structure of the proposed rigorous layer.

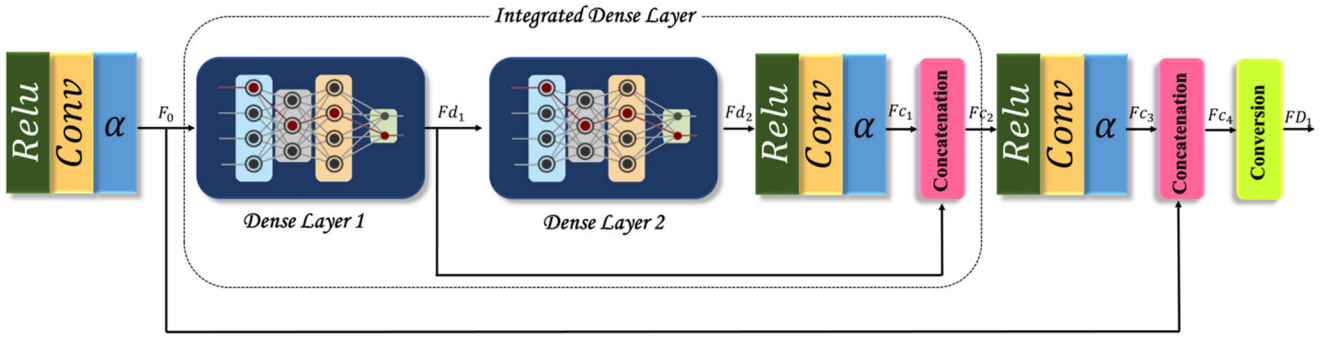


FIGURE 10. Structure of integrated dense layer within rigorous layer 1.

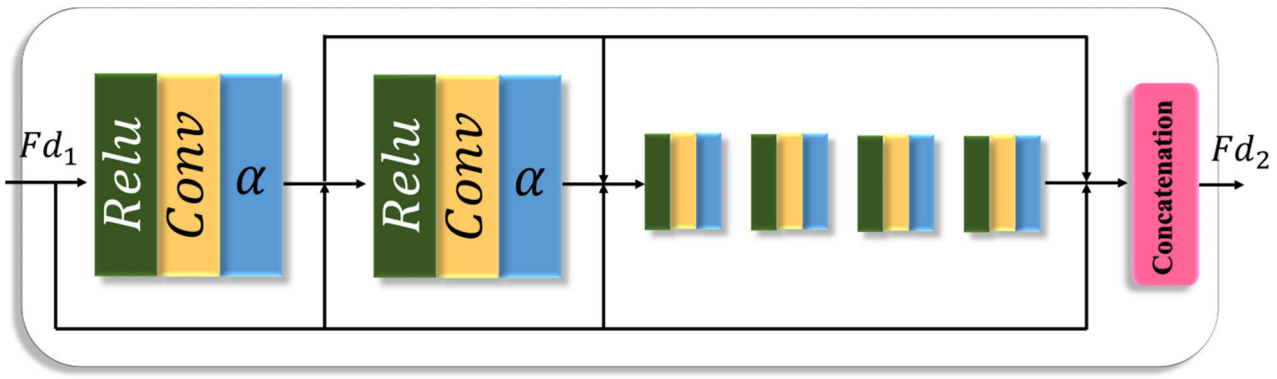


FIGURE 11. Structure of dense layer.

a: INTEGRATED DENSE LAYER

The integrated dense layer comprises two dense layers and a convolution process. F_0 is supplied into the first dense layer, as shown in Fig. 10, to produce dense features Fd_1 in that layer. Correspondingly, this work can extract dense features Fd_2 from the second dense layer is represented using Eq. (9).

In the Eq. (9), $H_{Dense2}(\cdot)$ and $H_{Concatenation}(\cdot)$ are functions of dense layer2 and concatenation, respectively, and for each layer of the dense layer, this work has $Fd_{1i}(i = 1, 2, \ldots c)$ inner layers; c denotes the total count of layers in dense layers (see Fig. 11). Feature Fd_2 has a channel count which is defined as $N_{Fd_2} = N_{Fd_1} + c * g$. where N_{Fd_1} represents the channel strength of features Fd_1, g is the growth rate. To get the features of Fc_1, this study performs a convolution operation on Fd_1. To improve dense layer mobility and fusion, this stage extracts and learns properties that are used in conjunction with properties from Fd_1.

b: DENSE INTEGRATION OPERATIONS

Dense Integration operations refer to the functions that remain in the rigorous layer after those that have already been completed. For combined features Fc_4, this paper performs a convolution operation on feature Fc_2 and add Fc_3 to the original input, the same as this study did in the Integrated Dense Layer. Fc_4 is finally down-sampled with the conversion layer. In Rigorous Layer 2, the conversion block serves

as the initial convolution and is a convolution operation. This work can connect the subsequent rigorous levels much more firmly this way.

As a result, this model fully utilizes the rigorous layer's internal feature information while increasing the agility and integration of global feature information across the entire framework. Using various convolution and concatenating processes to generate hierarchical shortcuts for different layers of features is one of this research's significant achievements. Because of this, the proposed network can thoroughly explore and utilize the various receptive fields [49]. Finally, this study's formula of a rigorous layer is presented in Eq. (10).

$$\begin{aligned} FD_1 &= H_{\text{convert}}\left(Fc_4\right) \\ &= H_{\text{convert}}\left(H_{\text{concat}}\left(Fc_3, F_0\right)\right) \\ &= H_{\text{convert}}\left(H_{\text{concat}}\left(H_{\text{conv}}\left(Fc_2\right), F_0\right)\right) \\ &= H_{\text{convert}}\left(H_{\text{concat}}\left(H_{\text{conv}}\left(H_{\text{concat}}\left(Fc_1, Fd_1\right)\right), F_0\right)\right) \\ &= H_{\text{convert}}\left(H_{\text{concat}}\right. \\ &\quad \times \left.\left(H_{\text{conv}}\left(H_{\text{concat}}\left(H_{\text{conv}}\left(Fd_2\right), Fd_1\right)\right), F_0\right)\right) \\ &= H_{\text{convert}}\left(H_{\text{concat}}\right. \\ &\quad \times \left.\left(H_{\text{conv}}\left(H_{\text{concat}}\left(H_{\text{conv}}\left(H_{\text{concat}}(A)\right), Fd_1\right)\right), F_0\right)\right) \end{aligned} \tag{10}$$

where $A = (Fd_1, Fd_{11}, Fd_{12}, \ldots, Fd_{1c})$ denotes an auxiliary matrix.

c: MANUSCRIPT LAYER

Predictions for each frame are utilized using the Manuscript layer to create the final label sequence, which consists of the Taylor-SoftMax and CTC. The predictions from the most recent dense block are generated using Taylor-SoftMax. As a result, CTC has a vital function in the final label sequence generated. The CTC takes each column of a picture with text sequence as data input and generates a set of characters as output.

In this proposed Rigorous layer, this model uses numerous shortcuts from IntensiveNet. Computer vision tasks requiring prominent levels of precision use IntensiveNet quite frequently. This approach aims to find solutions to image recognition's subproblems. This work lowers the number of parameters and computation costs by replacing the ineffective convolutional procedure with a very efficient depth separable convolution. To avoid losing feature information, this work does not utilize any pooling layers and instead down-sample using convolution and stride 2. As a result, BNFRNN is a true Fully-CNN. This work employs multiple coevolutionary and concatenating processes to connect the features from the input and dense layers, which may completely utilize the rigorous layer's inner features. This study uses the previous rigorous layer's transition layer as the starting point for the next rigorous layer's first convolution operation. Consequently, extracting and learning global aspects of the entire framework is possible. This paper can utilize the hierarchical characteristics of distinct dense layers and collect global features underutilized in IntensiveNet.

This model's first convolution includes several functions depending on the input. It seeks to extract shallow features from images not strongly represented in information. However, when confronted with highly dense photos, the initial convolution commonly sets its stride to 2 to reduce the original input's sample size. It will serve the same purpose as the transition layers in this scenario. However, the first convolution of the entire framework uses a larger receptive field by using a 5×5 kernel size rather than a 3×3 kernel size. It is possible to get higher results theoretically by using deeper networks and images with dense feature information. However, large-sized features must be down-sampled due to computational power and resource limitations. Upsampling, such as transpose convolution, can enlarge the original images. The outcome can be enhanced if we extend the network's depth more during mid-refinement.

IV. EXPERIMENT

A. DATASET

The performance of the proposed framework is assessed using the WebMarket dataset [51]. It consists of 3153 unlabeled images in the collection, all taken from within one-meter distance of shelves using three digital cameras. The images were taken naturally, with no additional lighting adjustments or viewpoint restrictions; however, most included the subjects' frontal views. Photos were gathered from 18 retail shelves, each measuring 30m long and having around six levels. There are three or four shelf levels in the range covered by each image. There are images of 100 different product classes in the dataset. Detailed product categories with slight differences in packaging are included in the dataset. The model can deal with packages because it was trained on such data. Each image is a jpeg with a resolution of 2592×1944 pixels and 2272×1704 pixels. The trained model can deal with damaged and multiple-item packages since high-resolution stores enough information for each object in the image.

B. IMPLEMENTATION DETAILS

The proposed framework was developed using the PyTorch framework. All the experiments conducted on the DELL Precision Tower 7810 workstation feature an Intel(R) Xeon(R) CPU E5-2620 v3 dual processor, 96 GB of RAM, and an NVIDIA Quadro K2200 graphics card. Table 3 shows the parameters and details of the proposed framework.

C. ABLATION STUDY

Images from WebMarket were divided into three groups: training, testing, and validating. 80% of the photos are used for training, 10% for validation, and 10% for testing. Images were chosen randomly to prevent the same shelf display from appearing in more than one of these subsets. If the intersection over the union between the recognized value and the ground

TABLE 3. Parameters of the proposed framework.

Parameters	Grocery Object Recognition
No. of epochs	100
Learning rate	1×10^{-2}
Train-Valid-Test split ratio	80-10-10
Batch size	16
Optimization algorithm	Adam
Loss function	Focal loss and Bounding Box regression loss
Activation function	Leaky ReLU and Sigmoid

truth value is more significant than 0.5, this paper considers the recognition correct. Accordingly, this work calculates Precision (the number of accurate detections over the total detections), Recall (the number of correctly detected goods over the goods visible in an image), and F-score (the harmonic mean of Precision and Recall) for each image. Fig. 12 shows the unrefined proposed model's performance. As the WebMarket Dataset's training data volume grows, the success rate stays close to 1 and does not fluctuate much in value over time. The failure rate is 85.1% at its lowest point. Thus, there is no discernible effect on the suggested model's performance from the correlation between training and test data. As the dataset grows, the unrefined model has a higher success rate.

Furthermore, updating and retraining training data is not required, which reduces costs overall. The unrefined model's results for Precision, Recall, and F-score are shown in Fig. 13. Increasing the relevant data quantity increases the Recall and F-score, but precision does not vary. Minor changes and high precision values indicate that the increase in relevant data has a reduced impact on precision. If the approach detects a region, it will likely be part of a product region. In comparison to recall, precision shows that certain products are missing. If grouping and refinement fail, they will appear as either undiscovered or incorrectly classified, as well as undetected or incorrectly detected items. Even if each step fails, the performance is reduced, but a workable recall is still achieved. The use of training data has a modest effect on recall. As the amount of relevant training data increases, the recall increases more slowly. The F-score rises in proportion to the amount of pertinent training data that may be linked to the rise in recall. This work may conclude that the proposed framework works well even if the product detection phase learning is not refined, but it can work even better if it is refined.

Our object detector, YOLOv5, identifies products and categorizes them into classes, such as chips, biscuits, pasta, etc. Following this, the text detector and the proposed BNFRNN network capture label information such as brand names, quantity, prices, etc. These results can be combined and displayed for both users and shopkeepers.

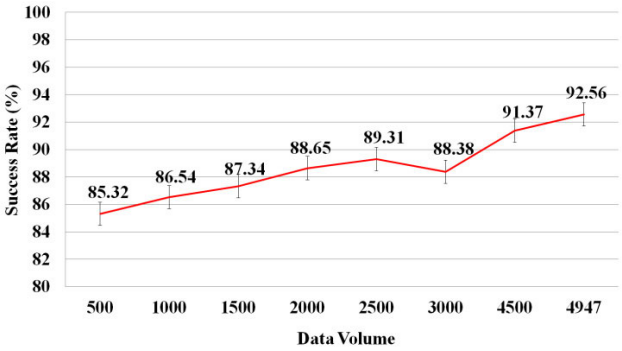


FIGURE 12. The proposed model's performance in terms of success rate.

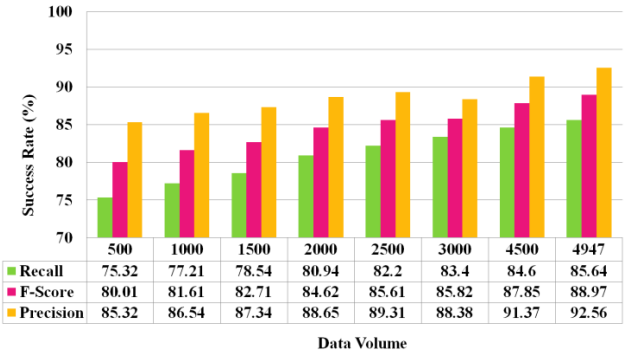


FIGURE 13. The proposed model's precision, recall, and f-score performance.

1) SIGNIFICANCE OF YOLOv5

We assess the performance of YOLOv5 with its earlier and later versions, such as YOLOv4, YOLOv7, and YOLOv8, in retail object recognition tasks (see Table 4). Compared to the YOLOv4 model, the YOLOv5 introduces new image augmentation techniques such as MixUp and CutMix, which mitigates the overfitting problem by generating a more comprehensive range of training samples, compelling the model to understand more generalizable features that remain consistent despite minor variations in the images. The proposed framework based on YOLOv5 outperforms YOLOv4 effortlessly by achieving an improved F-Score of +17.76%. On the other hand, the proposed framework combining YOLOv5 + Refiner + BNFRNN model achieves a better F-Score than the enhanced versions of YOLO by achieving an F-Score of (88.97% vs. 81.28% and 84.04%) for the custom dataset.

Table 4 shows that combining YOLOv5 with the refined block greatly helps the model perform better for custom object detection (grocery object detection). After a series of experiments, we found that the YOLOv5, YOLOv7, and YOLOv8 have showcased remarkable precision compared to earlier iterations of the YOLO model. Consequently, we also found that YOLOv7 (26.00 Frames Per Second (FPS)) and YOLOv8 (39.69 FPS) have a lower FPS than YOLOv5 (46.30 FPS). The generalized models must focus more on both accuracy and training time for real-world applications, whereas the YOLOv5 has direct attention on both parameters. The main reasons for selecting YOLOv5 as our object

TABLE 4. Performance comparison of YOLOv5 with other YOLO versions.

Models	WebMarket Dataset			
	Precision (%)	Recall (%)	F-Score (%)	Training Time (FPS)
YOLOv8	92.24	77.18	84.04	39.69
YOLOv5	89.00	78.00	83.14	**46.30**
YOLOv4	73.18	69.34	71.21	17.00
YOLOv7	84.55	78.25	81.28	26.00
YOLOv5 + OD-Refiner + BNFRNN	**92.56**	**85.64**	**88.97**	42.41

detector are its ease of training, reduced computational cost, faster inference speed, better results for custom datasets, and the ability to detect smaller objects. Additionally, the training time is lesser compared to other YOLO variants. Hence, we choose YOLOv5 as our object detector for retail product identification tasks.

The shared pooling feature in PANet and the multi-scale detection mechanism in the YOLOv5 algorithm efficiently handle oriented and skewed objects placed on shelves. Our primary challenge lies in our text recognizer's difficulty when predicting partially visible text. Nonetheless, our object detector has been crucial in achieving promising results in such scenarios.

2) SIGNIFICANCE OF OD-REFINER LAYER

Fig. 14 shows a numerical comparison of the evaluation metrics between the OD-refined and unrefined models. In contrast to the unrefined results, the refinement model shows improved Precision, Recall, and F-score scores. The recall has shown the most improvements. Refining the worst part of the system benefits the complete system more than just a specific product. The border can be extended to detect the product region that is near but not included in the vertical segmentation. The precision has also increased, indicating that the refining of object detection has successfully decreased the appearance of erroneous detection. Along with increased precision and recall, the F-score has improved as well. The results show that the refinements have impacted the proposed framework.

D. PERFORMANCE COMPARISON WITH BENCHMARK MODELS

The performance of the proposed model with other benchmark models (see Fig. 15); CNN-based descriptors such as [10] and [12] perform well, producing the highest Precision and F-score scores. Kumar et al. [52] produce comparable results, most notably recall. Additionally, it is worth

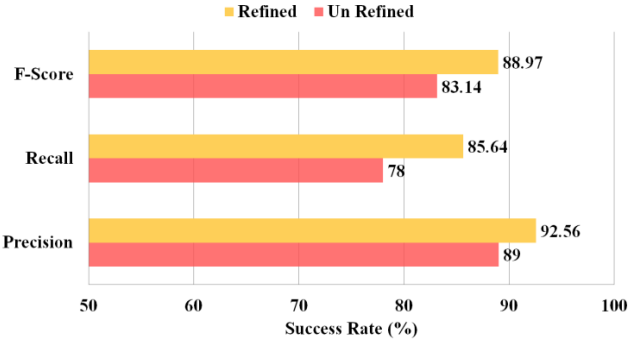


FIGURE 14. Performance evaluation between the OD-Refined and Unrefined model.

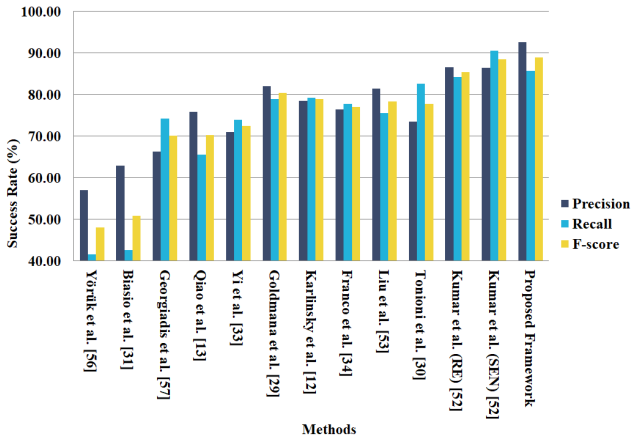
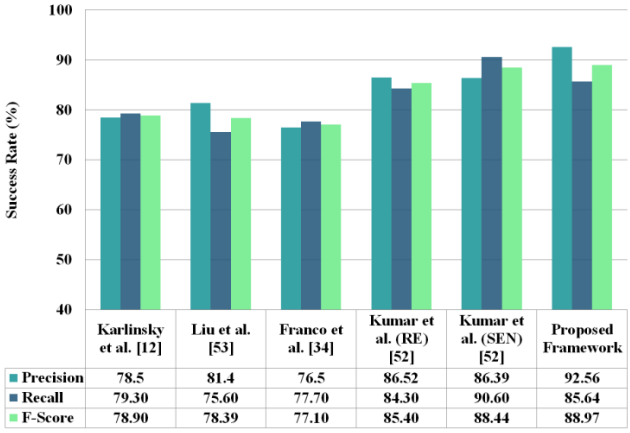


FIGURE 15. Performance evaluation between the proposed framework + without refinement layer vs. other benchmark models on the WebMarket Dataset.

noting how including numerous features, such as those proposed by Liu et al. [53], to capture distinct image structures may enhance the pipeline's sensitivity, as demonstrated by the most excellent recall. The OCR characteristics used by Karaoglu et al. [54] may achieve a comparable recall at the expense of precision. On the other hand, the use of colour descriptors does not appear to provide considerable benefits, which Tian et al's [55] technique ignored, resulting in competitive results.

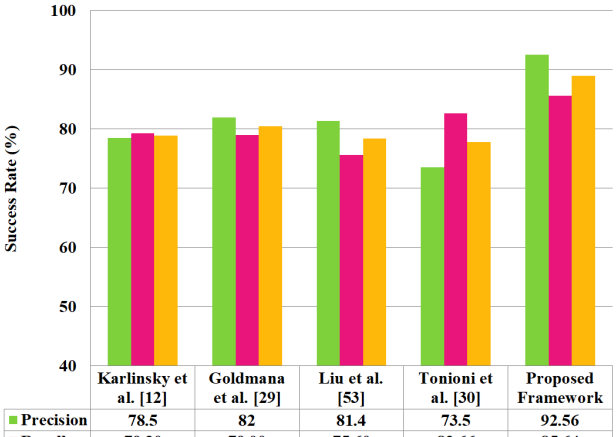
Given that the second phase is intended to eliminate false positives generated by the first, one would be tempted to select characteristics with a greater Recall. However, it may prove difficult for the second stage to identify objects with a problem if there are too many false positives. While the proposed model performed admirably, it falls slightly short of. Once this work incorporates the refinement phase of detection errors, a good balance between the models can be established. Therefore, this work will consider [11], [12], [33] [53], which performed well in the first round, in order to evaluate the results provided by the proposed pipeline following the refinement process further.

As illustrated in Fig. 16, once the proposed model incorporates the mid-refinement procedure, the model performs significantly better than the other models on all criteria.



	Karlinsky et al. [12]	Liu et al. [53]	Franco et al. [34]	Kumar et al. (RE) [52]	Kumar et al. (SEN) [52]	Proposed Framework
Precision	78.5	81.4	76.5	86.52	86.39	92.56
Recall	79.30	75.60	77.70	84.30	90.60	85.64
F-Score	78.90	78.39	77.10	85.40	88.44	88.97

FIGURE 16. Performance evaluation between the proposed framework + with refinement layer vs. other benchmark models on the WebMarket Dataset.



	Karlinsky et al. [12]	Goldmana et al. [29]	Liu et al. [53]	Tonioni et al. [30]	Proposed Framework
Precision	78.5	82	81.4	73.5	92.56
Recall	79.30	79.00	75.60	82.66	85.64
F-Score	78.90	80.47	78.39	77.81	88.97

FIGURE 17. Performance comparison between the proposed and OCR-based object recognition models.

Corrections for missing and erroneous BBs have demonstrated a more substantial influence on achieving more exact PR. In this example, the effectiveness of integrating the refinement step following the product detection phase can be argued. Finally, in Fig. 17, this research compares the effectiveness of this suggested model to two similar models developed by [12] [29], [30] and [53] in terms of OCR-based PR. The results indicate that this model retail product pipeline outperforms the other two algorithms across all experimented metrics. As a potential upgrade, the model can be learned to incorporate shape features to aid product recognition.

E. PERFORMANCE COMPARISON WITH EXISTING APPROACHES

Table 5 presents a comparative analysis of various methods' performance metrics, including precision, recall, and F-score, in a WebMarket dataset. The performance of the proposed framework is compared with various existing approaches. Biasio et al. [31], Yörük et al. [56], and Georgiadis et al. [57] exhibit moderate performance in Precision, Recall, and F-score, with values ranging from 48.10% to 70.11%. These methods identify individual objects but fail miserably to identify multiple retail objects. Karlinsky et al. [12], Qiao et al. [13], Goldmana et al. [29], Tonioni et al. [30], Yi et al. [33], Franco et al. [34], and Liu et al. [53] demonstrate higher performance, achieving F-scores between 70.32% to 80.47%. These methods struggle to identify dark objects, as well as small and medium-sized retail objects.

Notably, Kumar et al. (RE) [52] and Kumar et al. (SEN) [52] present robust results, scoring 85.40% and 88.44% in F-score, respectively. Santra et al. [38], Leo et al. [39], Gothai et al. [43], and Olóndriz et al. [40] present varying levels of performance, generally ranging between 63.84% to 71.00% in F-score. Selvam and Koilraj's method [42] stands out with high precision (89.40%), recall (88.20%), and an F-score of 86.30%. The existing approaches produce more false detections, resulting in more overlapping bounding boxes and difficulty identifying retail objects in complex backgrounds. The proposed framework exhibits the highest precision (92.56%) among all methods compared, with a recall of 85.64% and an F-score of 88.97%, showcasing notably strong performance across all three metrics. In our previous work [42], we introduced the Width-Height based Bounding Box Reconstruction algorithm, which helps the text detection model to capture the boundary characters. In contrast, we employed the state-of-the-art text recognition algorithm, SCATTER, to extract text information. However, conventional text on product packaging often features decorated characters, posing a challenge for SCATTER's performance. In this research work, the proposed BNFRNN network explicitly addresses the recognition of decorated and specially designed characters. Moreover, OD-Refiner conducts Redundant Bounding Box Removal operations, enhancing the object detector's performance, a feature absent in our prior research.

In summary, while several methods perform moderately well, Kumar et al.'s [52] approach, Selvam and Koilraj's [42] method, and notably, the proposed framework demonstrates superior performance in all three evaluation metrics such as precision, recall, and F-score, suggesting their potential effectiveness in the WebMarket dataset.

To demonstrate that the proposed method meets real-time requirements, we employed three important strategies: (i) Performance Metrics: We evaluated the performance of the proposed system using metrics such as precision, recall, F-score, and frames per second. Notably, the system achieved better scores in all these metrics. (ii) Benchmarking: A comparative analysis was conducted between the proposed system and state-of-the-art approaches developed between 2020 and 2023. The proposed system surpassed the majority of existing approaches, demonstrating a significant margin of accuracy. Notably, it exhibited superior real-time processing capabilities, particularly in handling complex benchmark datasets like WebMarket. (iii) Experimental Validation: We further validated the system's performance using real-world

TABLE 5. Performance comparisons of retail product detectors with existing methods on the WebMarket dataset.

Methods	Precision (%)	Recall (%)	F-score (%)
Karlinsky et al. [12]	78.50	79.30	78.90
Qiao et al. [13]	75.90	65.50	70.32
Goldmana et al. [29]	82.00	79.00	80.47
Tonioni et al. [30]	73.50	82.66	77.81
Biasio et al. [31]	62.90	42.60	50.80
Yi et al. [33]	71.00	74.00	72.47
Franco et al. [34]	76.50	77.70	77.10
Santra et al. [38]	70.40	68.40	69.40
Leo et al. [39]	66.30	70.30	68.20
Olóndriz et al. [40]	71.00	63.60	67.10
Selvam and Koilraj [42]	89.40	88.20	86.30
Gothai et al. [43]	58.00	71.00	63.84
Kumar et al. (RE) [52]	86.52	84.30	85.40
Kumar et al. (SEN) [52]	86.39	**90.60**	88.44
Liu et al. [53]	81.40	75.60	78.39
Yörük et al. [56]	57.00	41.60	48.10
Georgiadis et al. [57]	66.36	74.30	70.11
Proposed Framework	**92.56**	85.64	**88.97**

data, including grocery images from Google, Amazon, and Flipkart websites. The system demonstrated accurate detection and recognition of these images. We intend to deploy this system in an Android application.

F. COMPARISON OF TRAINING, CONVERGENCE, INFERENCE SPEED ACROSS MODELS

In this section, we conducted a comparison of the training time, inference time, and convergence speed of individual models (object detection, text detection, and text recognition) against an end-to-end framework. Tables 6 and 7 illustrate the training time and inference time, respectively, while Fig. 18 showcases the convergence speed of both individual models and the end-to-end framework.

From Table 3, it's noted that 80% of the dataset was utilized for training and the remaining 20% for validation and testing. It was observed that the object detector requires more training and inference time compared to the text detection and recognition models, given the specified hardware configuration. Conversely, the end-to-end framework demands more time for both training and inference when compared to stand-alone models.

From Fig. 18, it's evident that the convergence speed for text detection and recognition surpasses that of the object detection model. Specifically, convergence occurred by the 8^{th} epoch, with minimal deviation from the convergence point to the endpoint. In summary, the proposed text recognition model requires less training and inference time, demonstrating a superior convergence speed compared to both stand-alone and end-to-end frameworks.

TABLE 6. Performance comparison (Training Time) of individual models and the end-to-end framework. The training time is represented in the format hours:minutes:seconds (hh:mm:ss). The bold value indicates that the model took the least time for training compared to other models in the table.

Methods	Training Time (hh:mm:ss)
	WebMarket Dataset
Object Detection	02:11:31
Text Detection	01:43:23
Text Recognition	**01:28:12**
End-to-End Framework	04:24:06

TABLE 7. Performance comparison (Inference Time) of individual models and the end-to-end framework. The inference time is represented in the format hours:minutes:seconds (hh:mm:ss). The bold value indicates that the model took the least time for inference compared to other models in the table.

Methods	Inference Time (hh:mm:ss)
	WebMarket Dataset
Object Detection	00:25:04
Text Detection	00:18:22
Text Recognition	**00:14:19**
End-to-End Framework	00:47:34

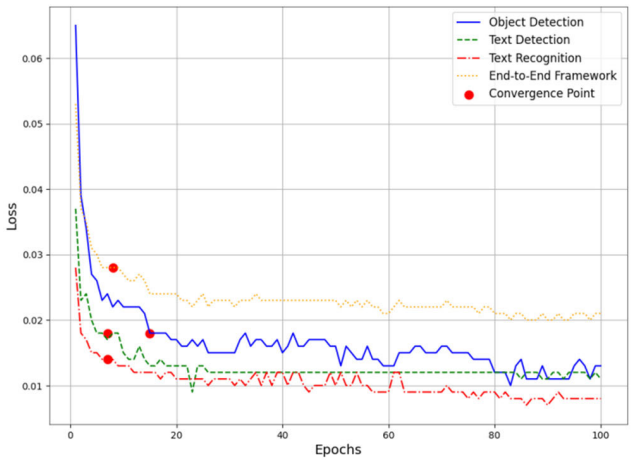


FIGURE 18. Visualization of convergence speed of the individual models and the end-to-end framework.

V. CONCLUSION AND FUTURE WORK

This work has provided a practical and straightforward solution to the problem of recognizing grocery products on store shelves in this study. The proposed framework utilizes

TABLE 8. List of symbols used in the research article.

Symbol	Description
b_j	Predicted bounding box
$\hat{b}_j$	Ground Truth bounding box
c	Objectness score
C_{max}	Maximum hypothesis score
F_n	The final feature map of the dense layer
FD_1	High-mobility features
g	Growth rate
H_{Comp1}	Comp layer1
$H_{FDL1}(\cdot)$	The function of Rigorous Layer 1.
$H_{FDL2}(\cdot)$	The function of Rigorous Layer 2.
$H_{Dense5}(\cdot)$	The function of the final dense layer.
$H_{Concatenation}(\cdot)$	Concatenation operation of dense layer
$H_{Conv}(\cdot)$	Convolutional operation of dense layer
$H_{Convert}(\cdot)$	Down-sample operation of dense layer
$H = \{\cdots h_i \cdots\}, h_i = \{n_I, n_O, c(n_I, n_O)\}$	Hypotheses
I	Reference planogram
IoU_j	Intersection over Union (IoU) for object j
$\mathcal{L}_{sIoU}$	Binary cross-entropy loss for Subtle-IoU layer
$\mathcal{L}_{Cl}$	Class loss
$\mathcal{L}_{Reg}$	Regression loss
$\mathcal{L}$	Overall loss of Subtle-IoU layer
O	Observed planogram
$\mathcal{O}^{iou}$	Subtle- Subtle-IoU score
N	Size of a grid
N_{Fd_1}	The channel strength of features Fd_1
$S \times S$	Size of image region
$tf.boolean_{mask}$	Boolean mask in Tensorflow
(α)	Learnable scalar multiplier

a three-stage pipeline to accomplish the task: (i) Object detection, (ii) Object refining, and (iii) Object recognition. As mentioned, the object detection phase uses the state-of-the-art object detection algorithm "YOLOv5". To overcome detection phase disparities, this paper introduced the "OD-Refiner" stage, which applies two refinement procedures: redundant box removal and missing box removal. This refinement phase is used to address the object detection model's shortcomings. The final stage of this work pipeline is object recognition. This work uses the OCR-based model "BNFRNN," a Fully-CNNs-based Feature Flow network that avoids the complexity of batch normalization. This phase determines the item's identity by parsing and matching the text against the database. The proposed model's precision, recall, and F-score are all evaluated. The findings are compared to those of other market leaders, and the proposed model demonstrates its efficacy by retaining a significant lead over its competitors.

In the future, we will strive to match the predicted product details with those stored in the database to address the complexity of predicting partially visible text information. In addition, we will try to improve the efficiency of retail object detection using an improved YOLOv8 algorithm and propose a new text recognition model based on a transformer-based network, which will help capture decorated and specially designed characters from the retail object package.

ACKNOWLEDGMENT

The authors are grateful to the Department of Computing, School of Technology and Innovations, University of Vaasa, Finland, for supporting this research work.

APPENDIX

LIST OF SYMBOLS

Table 8 describes the list of symbols used in the research article.

REFERENCES

[1] D. Grewal, A. L. Roggeveen, and J. Nordfält, "The future of retailing," *J. Retailing*, vol. 93, no. 1, pp. 1–6, Mar. 2017.

[2] X.-S. Wei, Q. Cui, L. Yang, P. Wang, and L. Liu, "RPC: A large-scale retail product checkout dataset," 2019, *arXiv:1901.07249*.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[4] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "DSSD : Deconvolutional single shot detector," 2017, *arXiv:1701.06659*.

[5] J. Redmon and A. Farhadi, "YOLOV3: An incremental improvement," 2018, *arXiv:1804.02767*.

[6] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1440–1448.

[7] Q. Zhao, T. Sheng, Y. Wang, Z. Tang, Y. Chen, L. Cai, and H. Ling, "M2Det: A single-shot object detector based on multi-level feature pyramid network," in *Proc. AAAI Conf. Art. Intel. (AAAI)*, vol. 33, Jan. 2019, pp. 9259–9266.

[8] F. Gu, J. Lu, C. Cai, Q. Zhu, and Z. Ju, "EANTrack: An efficient attention network for visual tracking," *IEEE Trans. Autom. Sci. Eng.*, vol. 7, no. 4, pp. 1–18, Oct. 2024, doi: 10.1109/tase.2023.3319676.

[9] F. Gu, J. Lu, C. Cai, Q. Zhu, and Z. Ju, "Repformer: A robust shared-encoder dual-pipeline transformer for visual tracking," *Neural Comput. Appl.*, vol. 35, no. 28, pp. 20581–20603, Oct. 2023.

[10] F. Gu, J. Lu, and C. Cai, "RPformer: A robust parallel transformer for visual tracking in complex scenes," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–14, 2022.

[11] A. Tonioni, E. Serra, and L. D. Stefano, "A deep learning pipeline for product recognition on store shelves," in *Proc. IEEE Int. Conf. Image Process., Appl. Syst. (IPAS)*, Dec. 2018, pp. 25–31.

[12] L. Karlinsky, J. Shtok, Y. Tzur, and A. Tzadok, "Fine-grained recognition of thousands of object categories with single-example training," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 965–974.

[13] S. Qiao, W. Shen, W. Qiu, C. Liu, and A. Yuille, "ScaleNet: Guiding object proposal generation in supermarkets and beyond," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 1809–1818.

[14] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.

[15] M. Merler, C. Galleguillos, and S. Belongie, "Recognizing groceries in situ using in vitro training data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.
[16] T. Winlock, E. Christiansen, and S. Belongie, "Toward real-time grocery detection for the visually impaired," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, San Francisco, CA, USA, Jun. 2010, pp. 49–56.
[17] G. Varol and R. S. Kuzu, "Toward retail product recognition on grocery shelves," in *SPIE Proc.*, Mar. 2015, pp. 1–7.
[18] M. Cotter, S. Advani, J. Sampson, K. Irick, and V. Narayanan, "A hardware accelerated multilevel visual classifier for embedded visual-assist systems," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Des. (ICCAD)*, Nov. 2014, pp. 96–100.
[19] D. Lopez-de-Ipiña, T. Lorido, and U. Lopez, "Indoor navigation and product recognition for blind people assisted shopping," in *Proc. Int. Workshop Ambient Assist. Living*, 2011, pp. 33–40.
[20] M. Kassim, C. K. H. C. K. Yahaya, M. H. M. Zaharuddin, and Z. A. Bakar, "A prototype of halal product recognition system," in *Proc. Int. Conf. Comput. Inf. Sci. (ICCIS)*, vol. 2, Jun. 2012, pp. 990–994.
[21] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jul. 2005, pp. 1–8.
[22] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 404–417.
[23] M. Marder, S. Harary, A. Ribak, Y. Tzur, S. Alpert, and A. Tzadok, "Using image analytics to monitor retail store shelves," *IBM J. Res. Develop.*, vol. 59, no. 2, pp. 31–311, Mar. 2015.
[24] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. CVPR*, Aug. 2001, pp. 511–518.
[25] V. Vapnik, S. E. Golowich, and A. Smola, "Support vector method for function approximation, regression estimation, and signal processing," in *Proc. 9th Int. Conf. Neural Inf. Process. Syst.*, Sep. 1996, pp. 281–287.
[26] M. George, D. Mircic, G. Sörös, C. Floerkemeier, and F. Mattern, "Fine-grained product class recognition for assisted shopping," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop (ICCVW)*, Santiago, Chile, Dec. 2015, pp. 546–554.
[27] A. Tonioni and L. D. Stefano, "Product recognition in store shelves as a sub-graph isomorphism problem," in *Proc. Int. Conf. Image Anal. Process., Lect. Notes Comput. Sci.*, vol. 10484, 2017, pp. 682–693.
[28] W. Geng, F. Han, J. Lin, L. Zhu, J. Bai, S. Wang, L. He, Q. Xiao, and Z. Lai, "Fine-grained grocery product recognition by one-shot learning," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 1706–1714.
[29] E. Goldman and J. Goldberger, "Large-scale classification of structured objects using a CRF with deep class embedding," 2017, *arXiv:1705.07420*.
[30] A. Tonioni and L. Di Stefano, "Domain invariant hierarchical embedding for grocery products recognition," *Comput. Vis. Image Understand.*, vol. 182, pp. 81–92, May 2019.
[31] A. D. Biasio and C. Fantozzi, "Retail shelf analytics through image processing and deep learning," M.S. thesis, Dept. Inf. Eng., Univ. Padua, Padua, Italy, 2019.
[32] S. Varadarajan and M. M. Srivastava, "Weakly supervised object localization on grocery shelves using simple FCN and synthetic dataset," in *Proc. 11th Indian Conf. Comput. Vis., Graph. Image Process.*, Hyderabad, India, Dec. 2018, pp. 1–7.
[33] W. Yi, Y. Sun, T. Ding, and S. He, "Detecting retail products in situ using CNN without human effort labeling," 2019, *arXiv:1904.09781*.
[34] A. Franco, D. Maltoni, and S. Papi, "Grocery product detection and recognition," *Expert Syst. Appl.*, vol. 81, pp. 163–176, Sep. 2017.
[35] S. Talha Bukhari, A. Wahab Amin, M. Abdullah Naveed, and M. Rzi Abbas, "ARC: A vision-based automatic retail checkout system," 2021, *arXiv:2104.02832*.
[36] G. Ciocca, P. Napoletano, and S. G. Locatelli, "Multi-task learning for supervised and unsupervised classifcation of grocery images. In: Pattern recognition," in *Proc. ICPR Int. Workshops Challenges*, vol. 12662, 2021, pp. 325–338.
[37] R. Yilmazer and D. Birant, "Shelf auditing based on image classification using semi-supervised deep learning to increase on-shelf availability in grocery stores," *Sensors*, vol. 21, no. 2, p. 327, Jan. 2021.
[38] B. Santra, U. Ghosh, and D. P. Mukherjee, "Graph-based modelling of superpixels for automatic identification of empty shelves in supermarkets," *Pattern Recognit.*, vol. 127, Jul. 2022, Art. no. 108627.
[39] M. Leo, P. Carcagnì, and C. Distante, "A systematic investigation on end-to-end deep recognition of grocery products in the wild," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 7234–7241.
[40] D. Amat Olóndriz, P. Palau Puigdevall, and A. Salvador Palau, "FooDI-ML: A large multi-language dataset of food, drinks and groceries images and descriptions," 2021, *arXiv:2110.02035*.
[41] A. D. L. Machado, K. Aires, R. Veras, and L. Britto Neto, "Grocery product recognition to aid visually impaired people," in *Proc. Anais Workshop de Visão Computacional*, Nov. 2021, pp. 94–99.
[42] P. Selvam and J. A. S. Koilraj, "A deep learning framework for grocery product detection and recognition," *Food Anal. Methods*, vol. 15, no. 12, pp. 3498–3522, Dec. 2022.
[43] E. Gothai, S. Bhatia, A. M. Alabdali, D. K. Sharma, B. R. Kondamudi, and P. Dadheech, "Design features of grocery product recognition using deep learning," *Intell. Autom. Soft Comput.*, vol. 34, no. 2, pp. 1231–1246, 2022.
[44] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang, "EAST: An efficient and accurate scene text detector," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2642–2651.
[45] Z. Zhang, Z. Tang, Z. Zhang, Y. Wang, J. Qin, and M. Wang, "Fully-convolutional intensive feature flow neural network for text recognition," 2019, *arXiv:1912.06446*.
[46] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2014, pp. 315–323.
[47] S. De and S. L. Smith, "Batch normalization biases residual blocks towards the identity function in deep networks," 2020, *arXiv:2002.10444*.
[48] S. R. Bulò, L. Porzi, and P. Kontschieder, "In-place activated BatchNorm for memory-optimized training of DNNs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5639–5647.
[49] I. Gitman and B. Ginsburg, "Comparison of batch normalization and weight normalization algorithms for the large-scale image classification," 2017, *arXiv:1709.08145*.
[50] P. Vincent, A. de Brébisson, and X. Bouthillier, "Efficient exact gradient update for training deep networks with very large sparse targets," 2014, *arXiv:1412.7091*.
[51] Y. Zhang, L. Wang, R. Hartley, and H. Li, "Where's the weet-bix?" in *Proc. Asian Conf. Comput. Vis.*, vol. 4843, 2017, pp. 800–810.
[52] M. Kumar, B. Moser, L. Fischer, and B. Freudenthaler, "Membership-mappings for data representation learning: Measure theoretic conceptualization," in *Database and Expert Systems Applications-DEXA*. Cham, Switzerland: Springer, 2021, pp. 127–137.
[53] L. Liu, B. Zhou, Z. Zou, S.-C. Yeh, and L. Zheng, "A smart unstaffed retail shop based on artificial intelligence and IoT," in *Proc. IEEE 23rd Int. Workshop Comput. Aided Model. Design Commun. Links Netw. (CAMAD)*, Sep. 2018, pp. 1–4.
[54] S. Karaoglu, B. Fernando, and A. Trémeau, "A novel algorithm for text detection and localization in natural scene images," in *Proc. Int. Conf. Digit. Image Comput. Techn. Appl.*, Dec. 2010, pp. 635–642.
[55] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao, "Detecting text in natural image with connectionist text proposal network," in *Proc. Eur. Conf. Comput. Vis. ECCV*, vol. 9912. Springer, 2016, pp. 56–72.
[56] E. Yörük, K. T. Öner, and C. B. Akgül, "An efficient Hough transform for multi-instance object recognition and pose estimation," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2016, pp. 1352–1357.
[57] K. Georgiadis, G. Kordopatis-Zilos, F. Kalaganis, P. Migkotzidis, E. Chatzilari, V. Panakidou, K. Pantouvakis, S. Tortopidis, S. Papadopoulos, S. Nikolopoulos, and I. Kompatsiaris, "Products-6K: A large-scale groceries product recognition dataset," in *Proc. 14th Pervas. Technol. Rel. Assistive Environments Conf.*, Jun. 2021, pp. 1–7.

PRABU SELVAM (Member, IEEE) received the B.E. degree in computer science from Shirdi Sai Engineering College, Bengaluru, the M.E. degree in computer science from Sathyabama University, Chennai, and the Ph.D. degree from SASTRA University, Thanjavur. He is currently an Assistant Professor with VIT University, Chennai. He has authored more than 20 research articles published in esteemed journals, such as Springer, Elsevier, and IEEE. His primary research interests include computer vision, deep learning, and pattern recognition.

MUHAMMAD FAHEEM (Member, IEEE) received the B.Sc. degree in computer engineering from the University College of Engineering and Technology, Bahauddin Zakariya University (BZU), Multan, Pakistan, in 2010, the M.S. degree in computer science from Universiti Teknologi Malaysia (UTM), Johor Bahru, Malaysia, in 2012, and the Ph.D. degree in computer science from the Faculty of Engineering, UTM, in 2021. In the past, he was a Lecturer with the COMSATS Institute of Information and Technology, Pakistan, from 2012 to 2014. In addition, he was an Assistant Professor with the Department of Computer Engineering, Abdullah Gul University (AGU), Kayseri, Turkey, from 2014 to 2022. He is currently a Researcher with the School of Computing Science (Innovations and Technology), University of Vaasa, Vaasa, Finland. He has numerous publications in journals and international conferences. His research interests include cybersecurity, Industry 4.0, smart cities, smart grids, and underwater sensor networks.

VIDYABHARATHI DAKSHINAMURTHI received the bachelor's and master's degrees in computer science and engineering from the University of Madras, Chennai, in 1998 and 2000, respectively, and the Ph.D. degree in information technology from Anna University, in 2023. She is currently an Associate Professor in computer science and engineering with the Sona College of Technology, Salem. Her research interests include machine learning, deep learning, data analytics, optimization, and data mining.

AKSHAJ NEVGI is currently pursuing the bachelor's degree in computer science and engineering with VIT University, Vellore, Tamil Nadu, India. He is a highly driven and accomplished undergraduate student with a passion for academic research. With a relentless commitment to knowledge and innovation, he has already submitted five research articles to reputable journals and presented his findings at the IEEE conference. His research interests include machine learning and deep learning and dedicated to making meaningful contributions to academia.

R. BHUVANESWARI received the Ph.D. degree from Anna University. She is currently an Assistant Professor with the Amrita School of Computing, Amrita Vishwa Vidyapeetham, Chennai, India. She has 18 years of teaching experience in the field of engineering. She has authored many publications in international journals and international conferences. She coauthored a book on computer graphics. Her research interests include machine learning and deep learning for image processing applications.

K. DEEPAK received the Ph.D. degree from Sastra University, Thanjavur. His Ph.D. thesis titled "Video anomaly detection using model-driven embeddings for visual events" has made significant contributions to the field. He is an accomplished researcher with over six years of experience in the fields of computer vision, deep machine learning, and speech and audio processing. He was a Postdoctoral Researcher with Université de Bourgogne. He is currently an Assistant Professor with Amrita Vishwa Vidyapeetham, Chennai Campus. In addition to his academic and postdoctoral roles, he actively contributes to the research community through his work at Amrita Vishwa Vidyapeetham, Chennai Campus. He has published numerous articles and research papers in renowned journals and conferences, demonstrating his dedication to advancing the frontiers of computer vision, deep machine learning, and speech and audio processing to address critical challenges in these domains.

JOSEPH ABRAHAM SUNDAR received the Ph.D. degree from SASTRA Deemed University, Thanjavur, in 2017. He is currently an Assistant Professor with the School of Computing, SASTRA Deemed University. With over 20 technical papers published and presented in international and national journals and conferences. His research interests include image processing and pattern recognition.