Journal of Theoretical and Applied Information Technology
30th June 2022. Vol.100. No 12

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ENHANCED SPATIAL PYRAMID POOLING AND INTERSECTION OVER UNION IN YOLOV4 FOR REAL-TIME GROCERY RECOGNITION SYSTEM

SAQIB JAMAL SYED[1], PUTRA SUMARI[1], HAILIZA KAMARULHAILI[1] ,VALLIAPPAN RAMAN[2] SUNDRESAN PERUMAL[2] ,WAN RAHIMAN[3]

[1]Research Scholar, UNIVERSITY SAINS MALAYSIA, School of Computer Science, Penang, Malaysia

[1]Professor, UNIVERSITY SAINS MALAYSIA, School of Computer Science, Penang, Malaysia

[1]Professor, UNIVERSITY SAINS MALAYSIA, School of Computer Science, Penang, Malaysia

[2]Professor, COIMBATORE INSTITUTE OF TECHNOLOGY, Department of AI and DS, India

[2]Associate Professor, UNIVERSITY SAINS ISLAM MALAYSIA, Faculty of Science and Technology, Malaysia

[3]Senior Lecturer, UNIVERSITY SAINS MALAYSIA, School of Electrical and Electronics Engineering , Malaysia

E-mail: [1]syedsaqib@student.usm.my, [1]putras@usm.my, [1]hailiza@usm.my, [2]valliappan@cit.edu.in, [2]sundresan@usim.edu.my,-[3]wanrahiman@usm.my

ABSTRACT

The ability to recognize a grocery on the shelf of a retail store is an ordinary human skill. Automatic detection of grocery on the shelf of retail store provides enhanced value-added to consumer experience, commercial benefits to retailers and efficient monitoring to domestic enforcement ministry. Compared to machine vision-based object recognition system, automatic detection of retail grocery in a store setting has lesser number of successful attempts. In this paper, we present an enhanced YOLOv4 for grocery detection and recognition. We enhanced through spatial pyramid pooling (SPP) and Intersection over union (IOU) components of YOLOv4 to be more accurate in making recognition and faster in the process. We carried an experiment using modified YOLOv4 algorithm to work with our new customized annotated dataset consist on 12000 images with 13 classes. The experiment result shows satisfactory detection compare to other similar works with mAP of 79.39, IoU threshold of 50%, accuracy of 82.83% and real time performance of 61 frames per second

Keywords: *Grocery Recognition, Yolov4, Object Localization, Deep Learning, Machine Vision*

1. INTRODUCTION

Grocery detection and recognition in retail stores have always been an interesting application. By detection, we refer to localization (scanning within image using box) of the object within the image and follow by naming (recognition) the object. The application helps retail store to generate an inventory of products available in the store at any point of time. It minimized the issue out of stock. For customer, it provides a value-added experience by reducing shopping time. For domestic ministry, it helps to consistently monitor retail price ceiling to all retail stores in a district.

Grocery detection and recognition is always a challenge task. The racks are typically cluttered and often not organized in a regular fashion. The use of different cameras resulting in different distributions of image intensities. The rack images are captured using handheld devices. This often results in image blur due to camera shake and jitter. The model should be designed in consideration of these issues and produce very accurate prediction. Since the introduction of convolutional neural networks, the detection frameworks have become increasingly fast

and accurate especially a recent real time models that can handle object detection and recognition quite well is YOLO v4. YOLOv4 has component called spatial pyramid pooling (SPP) that able to extract precise information of the input to create rich image feature map. The other component in YOLOv4 called Intersection over union (IOU) to localize the object within image. Both, the rich image feature map and localization improves the classification task. However, the SPP has the issue of fixed size vs different size of feature map and at the same time the IOU is still having quite large accuracy during bounding box-based localization.

In this paper we enhanced the component of YOLOv4 by introducing new Spatial pyramid pooling (SPP) component and new Intersection over union (IOU) component. Our proposed SPP able to detect small or distant objects in the image and create better feature map and proposed IOU to provide better bounding box-based localization in YOLOv4.

Our main contribution in this research is (i) new enhanced spatial pyramid pooling method. (ii) new enhanced intersection over union method. (iii) creating the custom fully annotated grocery dataset twelve thousand of grocery images (fish, vegetables, oil and etc) over 13 classes. The dataset reflects the local residential area and consider unique to others similar dataset.

The paper is organized as follows: Section 2 discussed about YOLOv4 background and literature review of similar work. Section 3 presented our dataset preparation and the proposed work. Section 4 elaborated experiment result. Finally, section 6 drew some conclusions and future works.

2. LITREATURE REVIEW

There are many object detections algorithm used in the literature such as RCNN, Fast-RCNN, Faster RCNN, SSD[[Liu et.al, 2016], RetinaNet[Lin et.al, 2017], RefineDetc[Zhang et.al, 2018], CenterNet[Zhou et.al, 2019] and FCOS[Tian et.al, 2019], YOLO variant and many more. In our work we focus on the latest YOLOv4 for grocery detection. The basic component of YOLOv4 is shown in figure 1. It consists of three main components, Backbone, Neck and Head. The Backbone uses CSPdarknet53 as main convolution process for producing image feature map. The CSPdarknet53 is a pretrained model using Imagenet dataset. The Neck simply improved the feature extraction map and send to Head component. The Head is composed of dense prediction and sparse prediction components to classify the object. The Neck uses Modified Spatial Pyramid Pooling (SPP) layer to get the large receptive field size beneficial in mapping maximum size input maintaining the fixed output (Bochkovskiy et al., 2020). The Head, Yolo v4 use the same Head as Yolo v3 which is used Intersection over union (IOU) component to perform dense prediction such as prediction of bounding box in relation to ground truth bounding box coordinates given by the user (Gong et al., 2020).

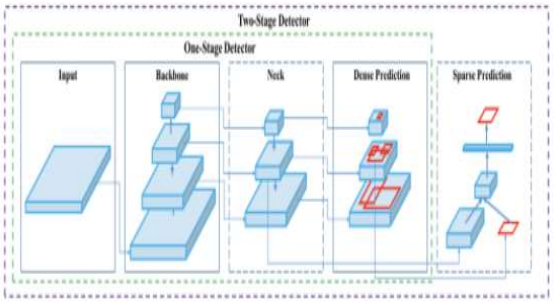


Figure 1: YOLOv4 basic architecture

Object detection requires algorithms to produce a series of bounding boxes with category scores, which can be roughly divided into two categories, i.e., anchor-based approach and anchor-free approach. The anchor-based approach uses the anchor boxes to generate object proposals, and then determines the accurate object regions and the corresponding class labels using convolutional networks. These categories are Faster R-CNN (Ren et al. 2017), Cascade R-CNN (Cai and Vasconcelos 2018). The anchor-free approach attracts much attention of researchers, including YOLO, CenterNet (Zhou, Wang, and Krahenb, 2019), FCOS (Tian et al. 2019) which generally produces the bounding boxes of objects by learning the features of several object key-points in real time. The anchor-free approach has shown great potential to surpass the anchor-based approach in terms of both accuracy and efficiency. The generic problem faced by these algorithms such as Fast-RCNN, Faster RCNN, and Yolov3 is these algorithms were low accuracy and low fps. It can only recognize the single object in a single frame due to small receptive field size. In Fast RCNN (Suhail et al., 2020), (Girshick, 2015) and Faster RCNN (Ren et al., 2017), Spatial Pyramid Pooling was introduced that can take multiple size images and generate the fixed size output. Although the accuracy was quite good as they totally rely on Region based Proposal Networks but they were quite slow in real-time maintaining the good fps rate. The problem of fps was solved by the W. Liu (Liu et al., 2016) in his Single Shot multi-box (SSD) algorithm.

Journal of Theoretical and Applied Information Technology
30th June 2022. Vol.100. No 12

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

He suggested two types SSD300 and SSD512 and performed the best result in real-time recognition. Another issue arose after he tested the algorithm and he faced that this algorithm failed to recognize the distant object that similarly happened in Redmon algorithm Yolo v3(Redmon & Farhadi, 2018). Although this algorithm showed optimal result in terms of overall accuracy than SSD but the speed was quite lesser. The only problem that was hard to achieve by these algorithms was the distant objects. Addie and his team [Addie Ira Borja Parico et.al, 2021] investigated on real time pear fruit counter using novel object detection model YOLOv4 and Deep Sort techniques for multiple object tracking. The results were not promising and accuracy achieved was less in obstacle environment. All these algorithms have achieved low accuracy in real time environment.

3. METHODOLOGY

We obtained large collection of custom grocery dataset, enhanced the YOLOv4 components specifically the SPP and IoU components and finally experimented the proposed work to assess the performance

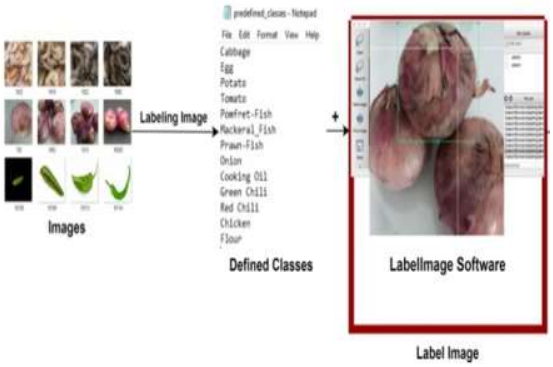
3.1 Customary Grocery Dataset Preparation

We created the dataset having 13 different grocery items (classes). The data collection was made manually from the supermarkets, night markets and different retail stores over residential area. To be able to examine the effect of the capturing quality, we capture photographs using iPhone 5S on its maximum resolution. Figure 2 shows the collected images from the supermarkets/stores. We later augmented (i.e. zooming, flipping and rotating etc) to achieve 12k images

Figure 2: Sample images of grocery dataset

To perform localization, we annotate all the 12k images one by one using Label-image software, so for every image a separate .txt file is created that contains the class label number as well as the coordinates of the ground truth bounding boxes as shown in figure 3. A separate names extension file is created for defining the class labels keeping the class names. The annotation files are defined in such a way that first number represent the class label while the rest of the four decimal numbers are x, y, w and h coordinates. Coordinates (x, y, w and h) inside the file highlights the location of the object enclosed in bounding box. Figure 3 shows the clear picture of creating the annotation files from the images.

Figure 4 clearly shows the concept of localization highlighting certain class labels as an example to cross check whether Pomfret-Fish, Tomato have class label number 3 and 4. Class label verifies the object identity as can be seen by arrow. Each arrow represents the class label number of respective class label following the other coordinates authenticate that this ground truth bounding box clearly validate the right class from the image for the object localization.



Class Label	Class	x	y	w	H
Pomfret-Fish	4	0.507	0.496	0.943	0.942
Tomato	3	0.236	0.590	0.260	0.225

Figure 3: Steps used in creating Dataset Annotation files

Journal of Theoretical and Applied Information Technology
30th June 2022. Vol.100. No 12

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

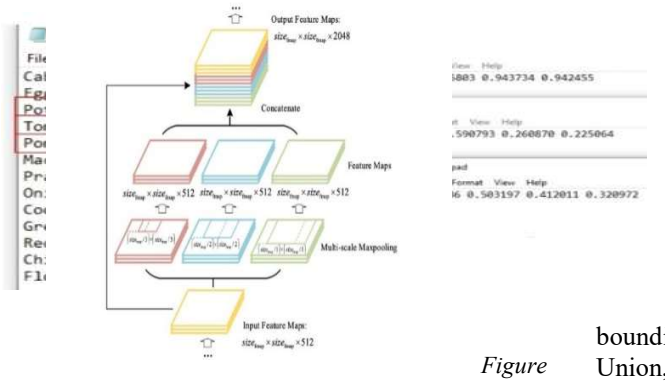


Figure 4: Localization

3.2 Enhanced Spatial Pyramid Polling

Many CNN-based models containing fully connected layers and therefore, accepts input images of specific dimensions only. The fully connected network requires the fixed size image, when detecting objects and most of the time the user don't have the fixed size images that's why it removes the part of the object we want to detect and therefore decrease the accuracy of our model. In contrast, SPP in the previous versions solved this problem and forces us to scale and accept the images having different sizes. At the output of the convolution neural networks, we have the features map. It will allow generating fixed size features whatever the size of our feature maps. To generate a fixed size output, it will use such as pooling layers like Max Pooling and generate different representations of our feature maps. Figure 5 (a) shows the 3 level bins i.e. 1,4 and 16 produces 3 vectors which later concatenated to form a fixed size vector which will be the input of the fully connected layer.

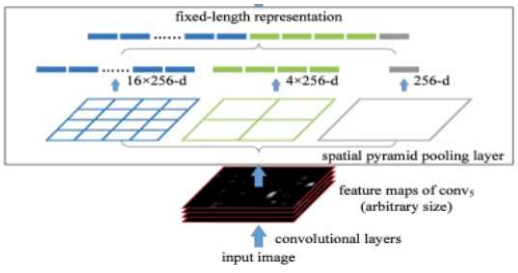


Figure 5: Spatial pyramid pooling component in YOLOv4

Figure 6. Enhanced Spatial pyramid pooling for YOLOv4

3.3 Enhanced Intersection Over Union(IOU)

Object Localization (Prediction through bounding box): IOU stands for Intersection over Union, a popular metric used to measure the accuracy of an object detector predicting the bounding boxes. Figure 7 shows YOLO divides the image into grids i.e. 9 grid cells. The yellow bounding box here is the ground truth box, the box you want to achieve so that object can be completely fitted into it with equal proportions from all sides while the black bounding box is the predicted one generated by the network during training (Gong et al., 2020)

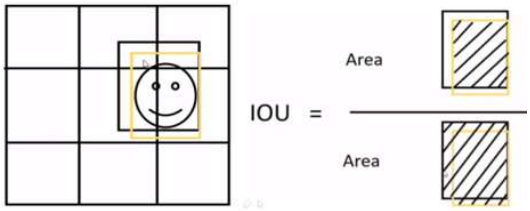
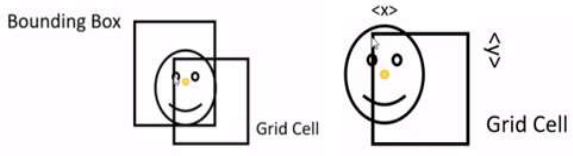


Figure 7: Original IOU method

In Intersection over Union, intersection is the common area taken from bounding boxes while the union is the overall area of both bounding boxes. Below is the formula of IoU (1).

$$IoU = \frac{Area\ of\ Intersection}{Area\ of\ Union} \quad (1)$$

From this formula, our goal is the good confidence score of each class, so that the overall probability shall be high. Figure 8 shows the concept of probability of object being an item in our case grocery.



Confidence Score [P(Object) * IOU]	X	Y	w	h	P(Grocery)

Figure 8: Calculating Probability of bounding box in

Journal of Theoretical and Applied Information Technology
30th June 2022. Vol.100. No 12

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

enhanced IOU method

Here x and y are the length of grid cell while w and h are the width and height of the bounding box. Our concern here is the probability of object being a grocery which is $P(Grocery)$ in our case grocery:

$$P(Grocery) = P\left(\frac{Grocery}{Object}\right) = Probability\ of\ Object\ being\ a\ Grocery \quad (2)$$

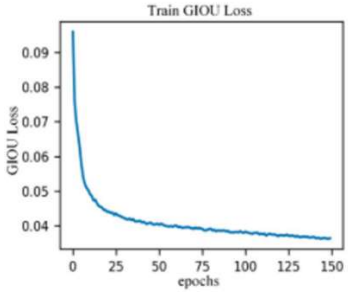
An Intersection over Union is considered to be "good" prediction if its confidence score > *0.5*. It can be seen in Figure 9.

Figure 9. Result of enhanced IOU method

Loss Function of Generalized IoU: It is a generalized Intersection over Union in which Ac represents the minimum bounding box between the predicted bounding box and the ground-truth bounding box. U is the union of the predicted and the ground-truth bounding boxes. The loss function not only pays attention to the overlapping area, but also focuses on the non-overlapping area (Gong et al., 2020). Below is the formula (3) and (4) for bounding box regression loss function used in this article. Figure 10 shows the graph of GIoU loss function we need to achieve. It decreases with increase in iterations.

$$GIoU = IOU - \frac{A_{c-U}}{|A_c|} \quad (3)$$

$$Loss_{GIOU} = 1 - GIoU \quad (4)$$



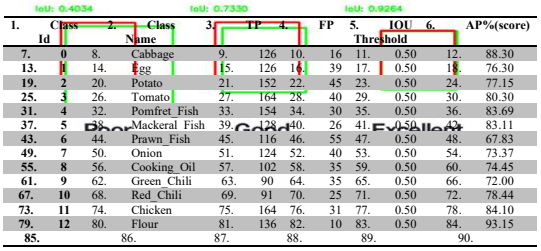


Figure 10: Graph of GIoU Loss Function

4 EXPERIMENTAL SETUP

The applications in this study were performed on NVIDA GeForce GTX 1080. More in detail, a special deep learning framework Darknet writen with C and CUDA has been used for YOLO [8]. A 30- layer architecture was created in order to solve the object detection problem in YOLOv2 by adding 11 layers to the 19 layered Darknet [9]. Darknet is open source, fast, easy to install and supports CPU/GPU calculations. In addition, the downloadable weights of YOLO can only be used in Darknet format. In this study, convolutional weights of YOLO, previously trained on Imagenet, were used for the training of object detection problem. The classes is 13 with learning rate 0.001. We split the dataset in the ratio 80:20 where 80 represent the percentage of training images while 20% for testing. Finally we give validation on 2.4k images out of 12k images. In order to get the better Mean Average Precision (mAP) and Accuracy, we tune the hyper parameters from the configuration file. In our experiments, the used batch value is 64 and the subdivision value is 16 for Grocery Dataset, the batch value is 32 and the subdivision value is 8 for ImageNet dataset in this study. The performance metrics as follows.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (5)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (6)$$

$$\text{mAP} = \frac{\text{I}}{\text{N}} \sum_{\text{i=0}}^{\text{N}} \text{APi} \quad (7)$$

$$\text{F1-Score} = \frac{2*\text{Precision}*\text{Recall}}{(\text{Precision+Rec})} \quad (8)$$

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+F}} \quad (9)$$

4.1.1 Average precision (AP)

Table 1 below shows the Average Precision (AP) value splitting all the classes to identify the

performance of each class. These evaluations are on the basis of 6910 recognition counts that identifies the unique truth count of 2432. Our model divided the unique truth count based on True Positive (TP) and False Negative (FN) while False Positive (FP) leads to the AP value of individual class tells us which class is performing well on the 50% of IOU threshold value of bounding boxes.

Table 1: Average Precision

4.1.2 Precision, recall and accuracy

True Positive (TP) is a correct recognition of predicted bounding box following the ground truth bounding box, False Positive (FP) is the misplaced recognition of an existing object. All the TP and FP values corresponding to each class is calculated by the model taken in the IoU Threshold of 50%. The values which are greater than IoU Threshold@0.5 means if the score of ground truth or actual bounding box overlapping with predicted bounding box is greater than 0.5 it lies in the True Positive otherwise, it considers as a False Positive. The Average IoU value generated by our model is 69.8% shows the percentage of the overall area common between the predicted and ground truth bounding box. The precision, recall and F1-Score as follows.

$Precision = 0.7966$ $Recall = 0.6879$

$F1\text{-}Score = \frac{2*Precision*Recall}{(Precision+Recal\ )} = 0.7382$

$Accuracy = 82.83\%$ $mAP = 79.39\%$

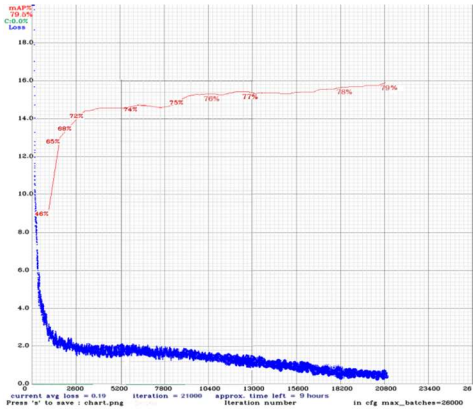


Figure 11. Box Loss

In Figure 11, it can be clearly seen that the loss is a box loss that depicts how good the center of an object is found and how good the predicted bounding box covers the object. Decreasing loss increase the mAP value of the model to localize the object well inside the bounding boxes. Here our average loss is 0.19 after 21000 iterations out of 26000 iterations and with that we get 79.39% mAP which is quite good for an object detector. We use Nvidia K80s as a GPU in Google Colab to train the model which is capable to work for not more than 12 hours a day so we ran model slowly and ended after it gets the sufficient mAP value.

4.1.3 Comparative Analysis with Existing works

Algorithm	Backbone (Pre-trained Dataset)	Size	FPS	Test mAP @50 (score)	Accuracy
Fast-RCNN	VOC2007, VOC2010 and VOC2012	512	6	68.4	78.82%
Faster-RCNN	VOC2007 and VOC2012	512	17	73.2	86.22%
SSD300	VOC2007 and VOC2012	300	59	74.9	73.41%
SSD512	VOC2007 and VOC2012	512	22	76.8	75.56%
Tiny Yolo-v3	Darknet-53 (ImageNet)	416	53	65.3	74.6%
Yolo v4 (Custom)	CSPDarknet-53 (ImageNet)	416	61	79.39	82.83%

For the experimental analysis, R. Girshik experimented on three different versions of Pascal VOC dataset starting from 2007 to 2012 and he found the best result in VOC 2012 up to 68.4 mAP score for Fast-RCNN *(Girshick, 2015)*. R. Dalai in his experiment for Faster-RCNN used same dataset with different classifiers such as SVM, Bayesian and CNN but found out the best accuracy using CNN method which is around 86.22% (Dalai & Senapati, 2017) looks like for making the accurate models Faster-RCNN can be used. W. Liu suggested about single shot detector algorithm training the model with Pascal VOC2012 dataset [9]. He used two different picture sizes i.e. 300 and 512 and found the good accuracy in both of them.

Table 2: Comparative Analysis

Later Redmon (Redmon & Farhadi, 2018) suggested Tiny Yolo v3 version trained on 20 classes using darknet-53 as a backbone which improved the mAP as well as accuracy of the model. Before Yolo v1 and v2 were trained on darknet-19 and because of that their mAP value was quite less than the latest

papers. Zhang suggested the comparison in his paper in which YOLO v3 used to be less in terms of mAP and Fps that YOLO v4 proved wrong (Chen et al., 2020). He suggested SSD to be accurate in terms of mAP and fast showing the maximum Fps value. Table 2 shows the comparison of the real-time objector detectors based on different algorithms in which we compared the image size almost onto the same scale for all the algorithms and we found out that Yolo v4 works quite well in terms of mAP and FPS than all of the algorithms but stay behind the Faster-RCNN algorithm in terms of accuracy. Figure 12 shows the FPS value of YOLO v4 in comparison with other algorithms. Here Yolo v4 is shown by YOLO (high) means Yolo v4 by far is the best when it comes to check Fps in real-time while second best in terms of accuracy.

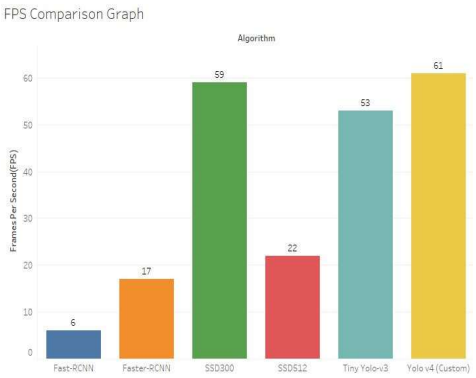


Figure 12: FPS comparison graph

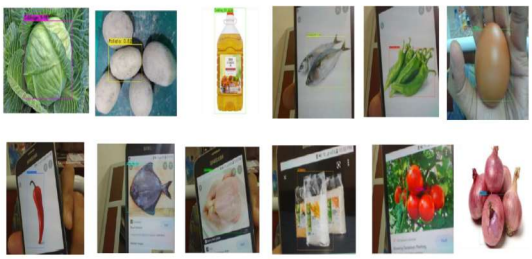


Figure 13: Set of result images

5 Discussion and Conclusion

In this research study, state of the art solution is spatial pyramid pooling (SPP) and Intersection over union (IOU) which offers higher accuracy of recognizing a grocery detection. However, the existing works are limited on evaluating grocery recognition with respect to it feasibility in uncertain scenarios, taking into the account of distribution of grocery products on the rack. Therefore, in this research, we investigated the enhanced YOLOv4, real time neural network detector model and demonstrated the detector which can be adapted to real time grocery detection. The results have achieved higher accuracy on par with the state of art solutions which is being highly efficient and it supports faster decision making.

In the conclusion, we successfully created a custom dataset by collecting images and drawing annotation for over 13 grocery items. We have successfully developed a model that is able to localize the objects well in real-time maintaining the good accuracy of small and distant objects. Finally our custom model can detect the objects with the mAP value of 79.39% while having overall model accuracy up to 82.83%. The most important thing we achieved is the speed of detecting the objects in real-time which approximately went to 61 frames per second which is quite better achieving results in real-time. In terms of accuracy, speed and computational cost for grocery recognition, enhanced YOLOv4 was found to be the most suitable, with an AP > 82% in real time environment. As a future work, some more products can be added and that should be handled in order to have an overall performance measure and a working prototype. Moreover, as a limitation, we have not applied any pre-processing such as noise reduction or perspective correction in this study. Such techniques will also allow us to improve our model accuracy in the future.

REFERENCES

[1] Liu, W, Anguelov, D., Erhan, D, Szegedy, C, Reed, S. E, Fu, C and Berg, A. C, "SSD: Single Shot MultiBox Detector", *In ECCV*, 21–37, 2016.

[2] Lin, T, Goyal, P, Girshick. R. B, He. K, and Dollar, P, "Focal Loss for Dense Object Detection. *In ICCV*, 2999–3007, 2017.

[3] Zhang, S, Wen. L, Bian. X, Lei, Z and Li. S. Z, "Single-Shot Refinement Neural Network for Object Detection", *In CVPR*, 4203–4212, 2018.

[4] Zhou. X, Wang. D and Krahenb, P, " Objects as Points". *CoRR abs/1904.07850*, 2019.

[5] Tian. Z, Shen. C, Chen. H and He, T, "FCOS: Fully Convolutional One-Stage Object Detection", *CoRR abs/1904.01355*, 2019.

[6] Athanasiadis, I, Mousouliotis, P and Petrou, L, " A Framework of Transfer Learning in Object Detection for Embedded Systems", 2018.

[7] Bochkovskiy. A, Wang. C.-Y and Liao. H.-Y. M, "YOLOv4: Optimal Speed and Accuracy of Object Detection", 2020.

[8] Borngrund, C, Bodin. U, and Sandin. F, "Machine vision for automation of earth-moving machines: Transfer learning experiments with YOLOv3", 2019.

[9] Chen. A, Yang. B, Cui. Y, Chen. Y, Zhang. S and Zhao. X, "Designing a supermarket service robot based on deep convolutional neural networks", *Symmetry*, 12(3), 2020.

[10] Dalai. R. and Senapati. K. K, "Comparison of Various RCNN techniques for Classification of Object from Image, *International Research Journal of Engineering and Technology(IRJET),* 4(7), 2017,

[11] Geethapriya. S, Duraimurugan. N and Chokkalingam. S. P, "Real time object detection with yolo", *International* Journal of Engineering and Advanced Technology, 8(3 Special Issue), 578–581, 2019.

[12] Girshick. R, "Fast R-CNN", *Proceedings of the IEEE International Conference on Computer Vision,* 1440–1448, 2015.

[13] Gong. B, Ergu. D, Cai, Y and Ma. B, "A Method for Wheat Head Detection Based on Yolov4. Research Square, 1–16, 2020.

[14] Liu. W, Anguelov. D, Erhan. D, Szegedy. C, Reed. S, Fu. C. Y and Berg. A. C, "SSD: Single shot multibox detector", Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics*)*, *9905 LNCS,* 21–37, 2016.

[15] Masurekar. O, Jadhav. O, Kulkarni. P and Patil, S, " Real Time Object Detection Using YOLOv3", *International Research Journal of Engineering and Technology (IRJET), 07*(03), 3764–3768, 2020.

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

[17] Ren. S, He. K, Girshick. R and Sun. J, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(6), 1137–1149, 2017.

[18] Suhail. A, Jayabalan, M and Thiruchelvam. V, "Convolutional neural network based object detection: A review", *Journal of Critical Reviews*, *7*(11), 786–792, 2020.

[19] Yun, Han. D, Chun. S, Oh. S. J, Choe. J and Yoo, Y, "CutMix: Regularization strategy to train strong classifiers with localizable features", *Proceedings of the IEEE International Conference on Computer Vision*, *2019-Octob*, 6022–6031.

[20] Addie Ira Borja Parico1 and Tofael Ahamed, "Real Time Pear Fruit Detection and Counting Using YOLOv4 Models and Deep SORT", *Sensors (Basel)*,21(14), 2021.