An Efficient Grocery Detection System Using HYOLO-NAS Deep Learning Model for Visually Impaired People

Chhabra, Payal
Dept. of Computer Science and Engineering, Maharishi Markandeshwar (Deemed to be University) M.M Engineering College

Goyal, Sonali
Dept. of Computer Science and Engineering, Maharishi Markandeshwar (Deemed to be University) M.M Engineering College

https://doi.org/10.5109/7236846

出版情報：Evergreen. 11 (3), pp.1990-2003, 2024-09. 九州大学グリーンテクノロジー研究教育センター
バージョン：

An Efficient Grocery Detection System Using HYOLO-NAS Deep Learning Model for Visually Impaired People

Payal Chhabra, Sonali Goyal*
Dept. of Computer Science and Engineering, Maharishi Markandeshwar (Deemed to be University)
M.M Engineering College, Mullana, Ambala, India.

*Author to whom correspondence should be addressed:
E-mail: payal49691@gmail.com

(Received April 16, 2024; Revised August 13, 2024; Accepted August 30, 2024).

Abstract: Real-time application of object detections are common, and the area of computer vision dramatically benefits from them. Recognizing grocery items poses a more significant challenge for blind individuals compared to those with normal vision. For that purpose, an effective model HYOLO-NAS, is used to detect groceries to aid the visually impaired by seamlessly converting text to audio messages. In the proposed work, Neural Architecture Search technology is used to dynamically update the weights that design child neural networks with the highest accuracy. The hyperparameter tuning on the child network involves adjusting the learning rate, number of epochs, and L2 regularization of weight decay with an Adaptive Moment Estimation optimizer. Google's Text-to-Speech (gTTS) transforms text into speech signals. After doing many inference experiments, the Hypertuned YOLO-NAS grocery detection model is introduced. The experimental results show that optimized HYOLO-NAS outperforms various detection algorithms with mAP0.5 reaching 96.80% on Grozi-120 and 97.61% on the Retail Product dataset.

Keywords: Deep Learning; Grocery Detection; YOLO-NAS; HyperParameter Tuning; Visually Impaired People.

1. Introduction

Object detection has achieved enormous advancements over time, driven primarily by deep learning techniques for blind people[36] due to their dependability on others for basic needs. The primary issue for them is buying their grocery products independently. Although various automatic product recognition has been widely used in supermarkets, grocery stores and the retail industry[1,31,15]. Still, one persistent and challenging issue for them is detecting and recognizing small-category grocery products such as soap, shampoo sachets etc. rather than large-category products such as bottles, boxes etc.[2,32].

Since 2015, the YOLO model detects objects which stands for You Only Look Once, and over the years there have been various improved versions up until YOLOv8 which was presented in 2023 by Ultralytics as shown in Fig.1. Now, in 2023 a newly released model called YOLO-NAS has launched[4]. Traditional model architectures are designed by human experts since there are many potential model architectures available[24,39], it is likely that even if it reaches great results, it takes times to create the best possible architecture manually.

Object detection techniques, like sliding window approaches, often struggle to cope with these challenges and might result in missed detections or false positive[29,30]. Modern methods, such as the YOLO family and SSD, have shown remarkable success, they still face considerable hurdles when dealing with small objects. YOLO-Neural Architecture Searches have been developed to identify small items that surpass the YOLO family. Compared with existing models, the Neural Architecture Search engine, AUTONAC creates a new YOLO-NAS model which acts as a baseline model in the proposed research. This model surpasses all other SOTA YOLOs in terms of speed and accuracy, including YOLOv5, YOLOv6, YOLOv7, and YOLOv8[5].

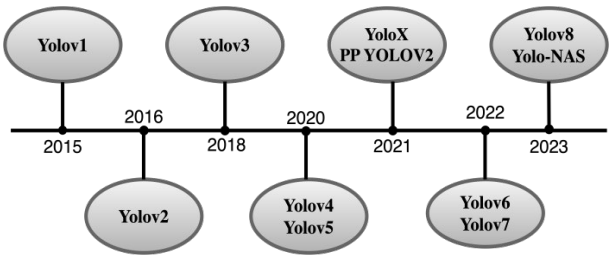

Fig 1: Timeline of YOLO Models from v1 to YOLO-NAS

1.1 Baseline Model

YOLO-NAS incorporates selective quantization and quantization-aware training, which use quantization to achieve maximum efficiency without compromising accuracy[6,40]. The model size is reduced after applying Quantization. There is very little precision loss when this model is converted to the INT8 version, a significantly improveming over other models. It uses neural search and is pretrained on the COCO dataset, making it suitable for small detection tasks. It has small, medium, and large versions. We implemented a small version of the YOLO-NAS model using different hyperparameters[6].

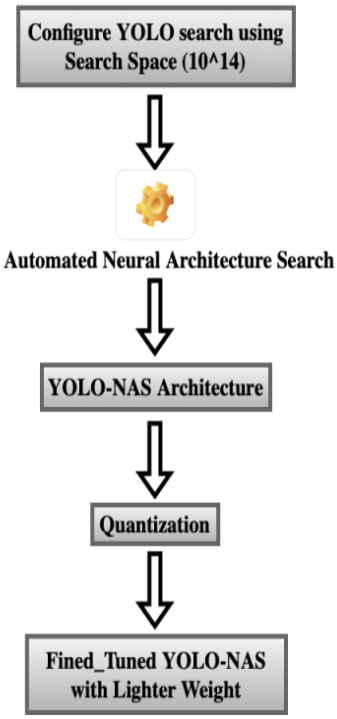

Fig 2: Depiction of Architecture search in YOLO-NAS.

YOLO-NAS introduced automated neural architecture search[21] to find an optimal and possible architecture, which creates an initial search space of huge size. Another attribute that helps this algorithm to have great results in super low latency is quantization, which reduces model weights to consume less memory and run faster. In Fig. 2, neural architecture search with its component is introduced[7,8].

1) The set of valid possible architectures to choose from Search space.
2) A search strategy used to dictate an algorithm to find possible optimal architectures from the search space.
3) An evaluation strategy is used to compare performance before training.

1.2 Problem Statement

This study aims to focus on the primary challenges faced by blind people in detecting grocery objects.

1) Many small items can be found in the images, which typically don't have enough appearance information. This commonly result in false positives and missed detections in detection tasks.

2) Small sizes of objects suffer from low resolution and reduced visual information, making them less distinct and more challenging to discriminate from the background.

Their limited spatial extent often leads to them being overshadowed by larger neighbouring objects, causing further confusion for the detection model. The inherent imbalance in the distribution of small objects versus larger ones in most datasets exacerbates the problem[3], leading to biased training and reduced generalization performance.

1.3 Primary Objectives: This study aims to accomplish the following goals.

(i) To gather grocery images of standard and small sizes under different conditions.

(ii) To determine which of the current CNN architectures is the optimal architecture model.

(iii) Use ADAM optimizer strategies along with CNN architects' hyperparameter adjustment to design a novel architecture for grocery detection.

In this proposed work, an audio-based HYOLO-NAS model with an ADAM optimizer[9] is implemented to detect groceries for blind individuals that will enable them to handle the daily grocery needs on their own. The model was trained and evaluated on Grozi-120[37] and Retail Product dataset[38] to identify grocery products.

2. Literature survey

Various deep learning techniques demonstrated remarkable performance in computer vision tasks. However, Detecting small objects remains a challenging problem due to their limited spatial extent, reduced visual information, and potential overshadowing by larger neighbouring objects. This literature survey covers three main areas: detection by YOLO-NAS, grocery detection, and advancements made in small object detection using various approaches, and methodologies proposed by researchers, as shown in Table 1.

Mengzi Hu et al.[10] apply spatial pyramid pooling to aerial pictures to improve small item visualisation. Additionally, a lightweight, efficient You Only Look Once technique is suggested to enhance small item recognition by using the α-complete IoU loss function, reducing differences between actual and predicted sample images of aerial datasets. In experiments, EL-YOLOv5 APs with 640 × 640 input size reached 10.8% and 10.7% and improved by 1.9% and 2.2% on DIOR and VisDrone datasets compared to YOLOv5.

Aduen Benjume et al.[11] evaluate how the YOLOv5 algorithm is modified to enhance its ability to identify smaller objects for self-driving cars. This is accomplished by introducing a new YOLO-Z model, which modifies each model's structure independently across all scales and treats them all as distinct models to assess how they affect performance and inference time. As a result, it increased mAP by 6.9% at 50% IoU from 0.869 to 0.925 and inference time 16.6 by 3 ms for smaller object detection compared to the original YOLOv5.

Hongxia Yu, Lijun Yun et al.[12] introduce the BigGhost module for remote sensing images, which optimizes the YOLOX to increase accuracy by fusing this model through modulated deformable convolution to improve small object performance. Simultaneously, the number of calculations and parameters is reduced to reduce inference time. Experimental findings demonstrate that BGD-YOLOX outperforms modern detection algorithms YOLOv4 in terms of average accuracy rate for small items, mAP0.5 and mAP0.95, reach 88.3% and 56.7%, respectively.

Shu Jun Ji et al.[13] proposed the multiscale contextual information YOLOv4 model to detect small-size car objects. It gather the feature context information that allows for the solution to this issue. By adding the attention module to PANet and implementing the EFB module at PANet and CSPDarknet53, the network was able to focus more on regions of interest rather than irrelevant data features. According to experimental results, 84.18% average precision (AP), recall 87.55% and the capability to comprehensively identify small car instances with 88.08% F1 score is achieved. Its effectiveness is further demonstrated on the Random Small Object Detection dataset, where it outperforms YOLO-X, v3, v4, v5, and RetinaNet with a mean average precision of 84.63%.

Prabu Selvam et al.[14] have introduced Width Height Bounding Box Reconstruction along with a backbone network (ResNet50 + FPN) to focus on detecting retail products. The grocery performance on the GroZi-120 achieves a Precision of 86.3%, 77.8% Recall and 77.04% F- Measure. On the web market, Precision is 89.4%, Recall is 88.2%, and F- Measure is 86.26%.

Gothai et al.[16] use YOLOV5 for grocery product detection and product recognition, it uses colour, shape and size features to reduce false product detection. Experimental results on the SKU 110K data set have improved precision by .11%, recall by .2%, and reduced Mean Average Error from 90.46 to 11.35%.

Tao Sun et al.[20] introduced the HPS-YOLOv7 model to detect small objects. During the feature fusion stage, the small object information is preserved using High Precision YOLOv7 for VisDrone2019 and Tinyperson datasets. 20.20 mAP increases by 3.0% compared to YOLOv7 for a tiny person. Replace the 20x20 detection head with a 160x160 detection head. A shallow feature fusion network is incorporated to retrieve information that is lost in the neural networks to capture and preserve details associated with smaller objects. Improves detection efficiency and accuracy during model training by utilizing a priori anchor adaptive adjustment strategy and mosaic data enhancement.

Zaipeng Xie et al.,[36] implement YOLO with ORB-SLAM(Simultaneous localization and mapping). Real-time detection with voice recognition system having 55.3 mAP and 35 FPS. Coordinate transformation can produce a dense navigation map that enables the user to plan a path and avoid global obstacle.

3. Performance Parameters

Performance metrics for evaluating the model's effectiveness in detection include the following parameters:

3.1 Mean Average Precision

A popular metric for model's accuracy that balances precision and recall at different confidence scores. mAP is expressed using the equation (1):

$$\mathrm{mAP} = \frac{1}{N}\sum_{i=1}^{N}\mathrm{AP}_i \tag{1}$$

Here,

N denotes the number of classes.

AP_i represents Average Precision for class i

3.2 Precision

It quantifies the fraction of instances that the model predicted as positive (correctly or incorrectly) out of all actually positive instances. Precision derived from equation (2):

$$\mathrm{Precision} = \frac{\mathrm{True\ Positive}}{\mathrm{True\ Positive} + \mathrm{False\ Positive}} \tag{2}$$

3.3 Recall

It represents the percentage of real positive cases that the model accurately detected. Recall is calculated by using equation (3):

$$\mathrm{Recall} = \frac{\mathrm{True\ Positive}}{\mathrm{True\ Positive} + \mathrm{False\ Negative}} \tag{3}$$

where False Negative which means the algorithm or technique does not predict the object, True Positive represents objects that are correctly predicted by the algorithm, False Positive means the model incorrectly predicts the object and True Negative denotes no prediction.

Table 1: Related Work on small object detection, Grocery detection and YOLO-NAS.

Author	Year	Framework	Backbone Network	Dataset	Improvements	Summary
Deci.[33]	2023	YOLO-NAS	Neural network architectures.	Roboflow-100.	The average mAP is .815. Increased as compared to other SOTA.	0.818 Average mAP for YOLO-NAS_M. 0.815 Average mAP for YOLO-NAS_S. mAP 0.801 for YOLOv8. mAP 0.734 for YOLOv7. mAP 0.676 for YOLOv5.
Kheireddine Choutri et.al[34]	2023	YOLO-NAS	Neural network architectures.	UAV's(Unmanned Aerial Vehicle) Images.	In comparison to YOLOv8, YOLONAS maintained its position as the top-performing model with 0.71 mAP and 0.68 F1 score.	The authors perform detection in two phases. The first phase uses fire, non-fire, and smoke images for fire detection, and geolocalization using UAVs. In the second, detection tasks were performed using YOLO.
Chenhao He[35]	2023	YOLO Family	YOLOv7, v6, v5, v8. and NAS with varying versions (small, medium, and big).	Outdoor Obstacles from the TT100K, COCO, and VOC datasets.	YOLOv7 exhibits the highest precision (0.786), recall (0.778), and 0.817 mAP, while YOLO-NAS-S demonstrates competitive precision (0.7888) but comparatively lower recall (0.5941) with mAP (0.6673).	Comparative Performance Analysis of different YOLO models. YOLOv5 with the least precision. YOLOv7 has the highest mAP and recall, with scores of 81.7% and 77.8%, respectively. With a recall of 59.41% and a mAP of 66.73%, YOLONAS-S exhibits the lowest recall and mAP.
Zhiwei Lin et al.[17]	2023	YOLO	HRNet for extracting features from small objects.	Self-constructed electric power operation scene dataset.	87.2% improved by 3.5 % as compared to YOLOv5.	Proposed small object HS-YOLO, algorithm based on High-Resolution Net. Additionally, Uses parallel branches and feature fusion for small object feature extraction.
Tanvir Ahmad et al.[18]	2020	YOLO v1	Inception module with convolution kernels of 1x1 to reduce weight parameters	Pascal VOC 2007 and 2012.	Achieves detection results of 65.6% and 58.7% Compared with YOLOv1.	Changed loss function by substituting the proportion style for the margin style. More adaptable and logical in terms of network error optimization. The Spatial Pyramid Pooling Layer has been added. Incorporates a 1x1 convolution kernel into the inception model and decreases the layers' total number of weight parameters.
ZhuangWang Et al.[19]	2021	Darknet53	Darknet53+ Denseblock is used for super-resolution strategy	Self-built dataset to detect small vehicles from aerial and satellite images.	89.91% mAP. Increase by 2.37% compared to YOLOv4.	FTT module employed for eliminating input image noise. The backbone network's feature extraction is optimized by discarding CSPDarknet53's CSP portion with dense block connections that reduces parameters and enhancing accuracy. Enhance image resolution, Combine SPPnet and PANnet in the neck for multi-scale feature fusion.

3.4 F1-Score

It provides a single value that summarizing the overall effectiveness of precision and recall.The F1-Score, which uses harmonic mean of recall and precision is computed by equation (4):

$$\text{F1-Score} = \frac{2*(\text{Precision}*\text{Recall})}{(\text{Precision} + \text{Recall})} \tag{4}$$

3.5 Intersection over Union (IoU)

The area of the intersection divided by the union area is known as the intersection over the union. The equation (5) is the mathematical expression for calculating the IoU score:

$$\text{IoU} = \frac{\text{Area of Intersection}}{\text{Area of Union}} \tag{5}$$

Let B1 and B2 be the name of two bounding boxes. The top left coordinates of box B1 are (X_{B1min}, Y_{B1min}), and the coordinates of the bottom right corner are (X_{B1max}, Y_{B1max}).Similarly, for box B2, the coordinates are (X_{B2min}, Y_{B2min}) and (X_{B2max}, Y_{B2max}).

The area of boxes B1 and B2 is given by equations (6) and (7).

$$\text{Area}_{B1} = ((x_{B1max} - x_{B1min}) * (y_{B1max} - y_{B1min})) \tag{6}$$

$$\text{Area}_{B2} = ((x_{B2max} - x_{B2min}) * (y_{B2max} - y_{B2min})) \tag{7}$$

The coordinates of the intersection box I are:

X_{Imin}= max(X_{B1min} , X_{B2min})
Y_{Imin}= max(Y_{B1min} , Y_{B2min})
X_{Imax}= min(X_{B1max} , X_{B2max})
Y_{Imax}= min(Y_{B1max} , Y_{B2max})

The area of the intersection box I is given in equation (8)

$$\text{Area}_I = \max(0, X_{Imax} - X_{Imin}) * \max(0, Y_{Imax} - Y_{Imin}) \tag{8}$$

Now, Intersection over Union is calculated as:

$$\text{IoU} = \text{Area}_I/(\text{Area}_{B1} + \text{Area}_{B2} - \text{Area}_I) \tag{9}$$

4. Proposed Architecture

The YOLO series has made significant progress but still has a high miss detection rate when it comes to small product identification because of their low resolution and lack of knowledge about their attributes. The model's structure and the properties of small items are the primary factors in detecting small objects. So, the Hyperparameter tuning of the YOLO-NAS model is done to get more precise information about the attributes of small items.

In the first Phase, an ADAM optimizer with 0.0001 L2 regularization weight decay is used, which encourages the model to maintain smaller weights[18], which is essential in tasks where the risk of overfitting is high. Parameters are given by equation (10):

$$W_{t+1} = W_t - \frac{\text{learning rate}}{\sqrt{v_t} + e} . \left(\frac{m_t}{\sqrt{v_t} + e} + \text{weight\_decay}. w_t\right) \tag{10}$$

where,
w_t parameter at time t,
m moving average of gradients,
$\sqrt{v_t}$ moving average of squared gradients,
e is constant to avoid zero division.

In the Second phase, hyperparameter tuning is performed to address grocery products.

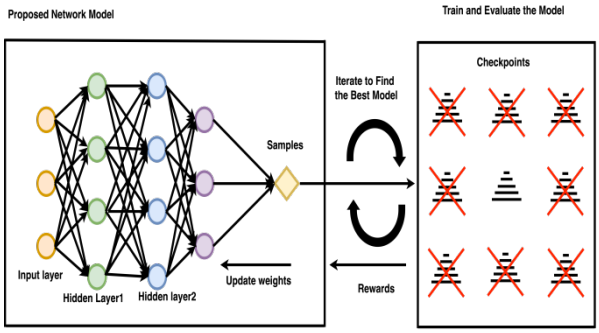

Fig. 3: Architecture of the proposed optimized HYOLO-NAS

In Fig. 3, below mentioned parameters are given to the input layer, which determine the network depth and the training's duration[46]. An optimal child network is created by passing block size, number of epochs, and learning rate that will further train on the training and validating dataset then will save the achieved accuracy as a checkpoint. A Checkpoint is a saved model file that contains the parameter values of a neural network at a specific point during training. Accuracy will pass as a reward to the network, accordingly, it will update the weight. So, the next iteration or time stamp the network will use those actions that will give a better network configuration.

The model focus on hyperparameter tuning and the use of the ADAM optimizer with L2 regularization weight decay which specifically enhance the precision of small object detection, addressing a known limitation of the YOLO series. The ADAM optimizer provides an adaptive learning rate, and L2 regularization encourages smaller weights, which can help prevent overfitting. It leads to more efficient and effective training, particularly when detecting small grocery objects. Additionally, the second phase's fine-tuning of hyperparameters like block size, number of epochs, and learning rate allows for the customization of the model to better handle the grocery product detection. Checkpoints ensures that progress is not lost, and the best performing model configurations are retained, saving time and resources. Reward-based system for updating weights encourages the model to continuously improve its performance. However, The two-phase approach, including the adaptive optimization and fine-tuning processes, can significantly increase the overall training time.

4.1 Learning rate

During training, the learning rate dictates how frequently the neural network's weights are updated.

Learning rate is represented by equation (11):

$$a_t = a_0 + m.t \quad (11)$$

Where,

a_t represents learning rate at iteration t
a_0 represents initial learning rate
m represents the rate of increase in the learning rate
t is the total number of epochs

4.2 Batch Size and Epochs

The number of samples handled during training is called batch size. Epochs describe a single pass across the training dataset which is used for training neural network.

4.3 Algorithm

```
1: Define Search Space for Hyperparameters
2:   hyperparameter_space =
{
        block_size = 16
        num_epochs= 20
        learning_rate = 0.000001
}
3: Initialize the Current Best Architecture and Accuracy
    {
        best_architecture = None
        best_accuracy      = 0.0
    }
4:  Hyperparameter Tuning and NAS
{
        for iteration in range(num-iterations):
                i. generate-random-
                   hyperparameters(hyperparameter-
   space)
        Build Neural Network with Hyperparameters:
                  i. child-network=build-neural-
                network(hyperparameters)
        Train the Child Network on Training Dataset:
        {
                  i. train-dataset,val-dataset=load-
                datasets()
                  ii. child-network=train-neural-
                network(child-network, train dataset)
                  iii. hyperparameters(['num-epoch'],
                hyperparameters['learning-rate'])
         }
    }
5:  Evaluate Child Network on the Validation Dataset
        i. accuracy = evaluate-neural-
                network(child-network, val-
                dataset)
6: Save Checkpoint if Accuracy Improved
   {
        if accuracy > best-accuracy:
                best_accuracy = accuracy
                save-checkpoint(child-network,
                hyperparameters, accuracy)
   }
7:  Update Parent Network's Parameters (Weights) based on Accuracy
                i. Update-parent-network parameters
                (hyperparameters, accuracy)
8: Best architecture is saved in checkpoint file with highest accuracy.
9: Load the pre-trained model, and do a performance analysis
10:  Convert text recognized by detection into audio for the visually impaired.
```

The algorithm begins by defining a search space for hyperparameters, specifying values such as block size, number of epochs, and learning rate. This initial setup provides the parameters to be adjusted during the training process. The next step is initializing variables to track the best architecture and highest accuracy achieved. The algorithm then enters a hyperparameter tuning and neural architecture search (NAS) phase, where it iterates through a predetermined number of cycles. In each iteration, it generates random hyperparameters from the defined search space and uses them to build a neural network, called the child network. This child network is trained on the training dataset with the specified epochs and learning rate. Use a validation dataset to test the child network's accuracy after training. The current model's state is saved as a checkpoint if it surpasses the previously recorded best accuracy. The parent network's parameters are then updated based on the achieved accuracy, ensuring iterative improvement. The architecture that achieves the highest accuracy is saved in a checkpoint file. Subsequently, the best performing model is loaded for performance analysis. Finally, the recognized text from the detection process is converted into audio, enhancing accessibility for visually impaired individuals. This structured approach ensures continuous optimization and effective neural network, performance, particularly in detecting small objects.

4.4 Flowchart

Figure 4 shows the flow of grocery detection from creating an optimal and best network using fine-tuning hyperparameters by saving the best result in the checkpoint directory to converting it to audio[44]. This framework is designed to optimize the inference speed and accuracy of deep neural networks, while keeping its original accuracy as a baseline.

5. Experimental Setting

This section discusses the experiments in detail. The entire experimental procedure has three distinct phases. First, provide details of the dataset. Second, experimental details, the hyperparameter setting of the model, and the working platform on grocery images. Third, introduce the image evaluation result.

5.1 Dataset

Datasets have been crucial throughout the evolution of object detection research. They serves as a benchmark for evaluating algorithms and progress towards more complicated detection scenarios. In the proposed work, an object detection task is performed on two datasets including Grozi-120 and Retail Product.

5.1.1 Grozi-120

It contains 120 product categories, containing a total of 4829 mages. There are 3255 images in the training set, 1043 in th valid set and 531 in the test set[38].

5.1.2 Retail Product

Images used for training and testing of small-size grocery datasets with 2611 images[37]. 20 classes of product, using 2298 images for training, 190 valid sets and 123 test sets.

5.2 Experimental Details

For training, 640*640 fixed image size of the Grozi-120 and Retail Product datasets is used. The backbone network is a neural architecture search. Considering the system's configuration, there are 16 batches, 1e-6 warmup initial learning rate, and 2e-4 initial learning rate, which is adjusted using cosine value 0.1.

5.3 Hyper Parameter Tuning

Hyperparameter tuning involves different values of parameters like batch size, learning rate and epochs under each configuration to achieve the best possible performance from the object detection model[27,28,47]. The HYOLO-NAS model uses Table 2 parameters to train neural networks with given values, presumably for a grocery detection task.

5.3.1 Learning Rate

Linear warm-up with Cosine Annealing steadily increases the learning rate in linear steps during the initial training epochs. The 0.000001 Learning rate allows the model to converge to very fine-grained details in the data, which is beneficial when small objects demand high precision. After the warm-up, the initial learning rate for the main training phase is 0.0002.

Table 2: Parameters Tuning of Proposed HYOLO- NAS Framework

Parameters	**Object Detection**
Learning rate	$1*10^{-6}$
Optimization Algorithm	ADAM with weight decay
Learning Rate Warm-up Epochs	3
Batch size	16 with 2 threads
No. of epochs	20
Input Size	640*640
Loss function	PPYoloELoss

5.3.2 Batch Size

In the HYOLO-NAS model, 16 batches and 2 parallel processes or threads are processed simultaneously to load the data. A higher number of workers can lead to faster data loading, especially when dealing with large datasets.

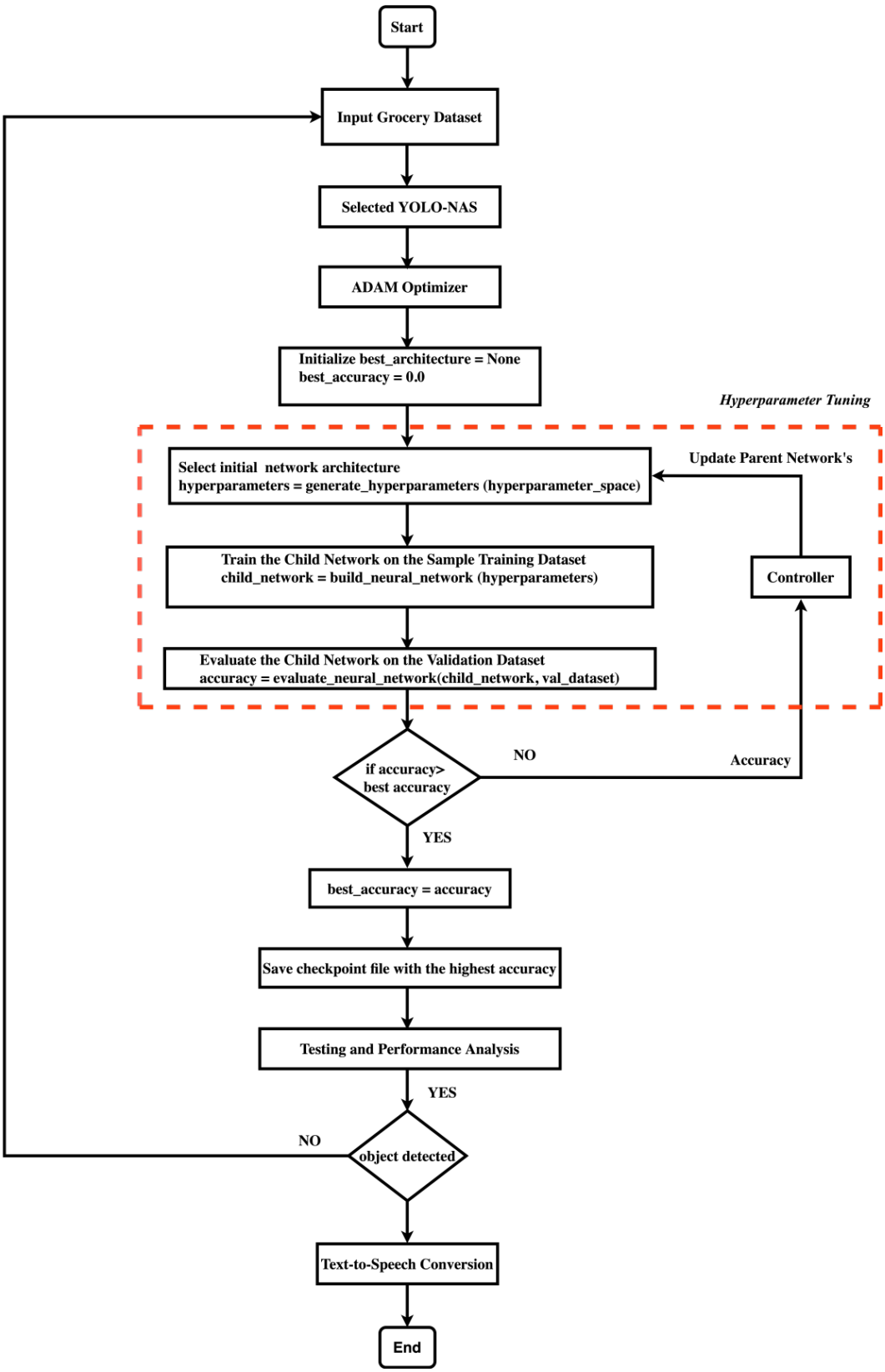

Fig. 4: Flowchart of Grocery Detection using HYOLO-NAS model

5.3.3 Epochs

The standard procedure begins with a reasonable value of 10 epochs as a baseline in HYOLO-NAS. This neural network model runs through 20 epochs, processing the complete dataset, computing gradients, and updating the model's parameters at each epoch.

5.3.4 Adaptive Moment Estimation Optimizer

ADAM implemented with 0.0001 L2 regularization weight decay encourages the model to maintain smaller weights, which is essential in tasks with high risk overfitting. Weight decay is a regularization technique that prevents overfitting, especially in tasks where accurate small object detection is a primary objective[22,23]. Training deep neural networks for grocery detection is a delicate process, and can cause overfitting especially when dealing with limited data of small objects. So, a moderate amount of L2 regularization or weight decay of 0.0001 is applied during training, which encourages the model to have smaller weights and can provide stable updates during training.

5.3.5 Non-max Suppression

To evaluate and test the object detection model's performance. The following threshold values are taken[24,25,26]:

i. score-threshold=0.01: Sets a score threshold for post-processing predictions.

ii. nms-top-k=1000: Specifies the maximum number of predictions to keep after non-maximum suppression (NMS).

iii. max-predictions=300: Sets the maximum number of predictions to retain after post-processing.

iv. NMS threshold: Set 0.7 threshold for non-maximum suppression.

6. Results

The findings on grocery detection are discussed in this section. The experiments were conducted on a diverse and representative Grozi- 120 dataset and a Retail Product dataset. Table 3 shows that the proposed HYOLO-NAS model outperforms several other methods on Grozi-120. Geng et al.[42] explored different configurations with VGG16, achieving a Precision of 50.44%, Recall of 30.69%, and a mAP of 63.17%. The extensions of VGG16 with attention maps based on SIFT (49.05% Precision, 29.37% Recall, and 65.55% mAP) and BRISK (46.32% Precision, 29.50% Recall) showed varied results. Bikash Santra et. al[43] achieved a notable mAP of 84.58%. However, Prabu Selvam et. al[44] demonstrated robust performance with an 86.3% Precision, 77.8% Recall, and 72.4% F1 Score on the Faster RCNN model, showcasing the effectiveness of their method on the Grozi-120 dataset. Leonid Karlinsky et. al[41] achieved an mAP of 49.8% with the YOLOv5 model, but the proposed HYOLO-NAS model outperformed several methods with an impressive Precision of 82.3%, 92.1% Recall, 96.8% mAP, and 84% F1 Score.

Table 3: Performance comparison of various approaches on Grozi-120 datasets

Grozi -120				
Methods	**Precision**	**Recall**	**mAP**	**F1 Score**
VGG16[42]	50.44%	30.69%	63.17%	-
VGG16+ATmap SIFT[42]	49.05%	29.37%	65.55%	-
VGG16+ATmap BRISK[42]	46.32%	29.50%	-	-
VGG-19[43]	-	-	84.58%	-
YOLOv5[44]	86.3%	77.8%	-	72.4%
FRCNN[41]	-	-	49.8%	-
HYOLO-NAS(our proposed)	82.3%	92.1%	96.8%	84.0%

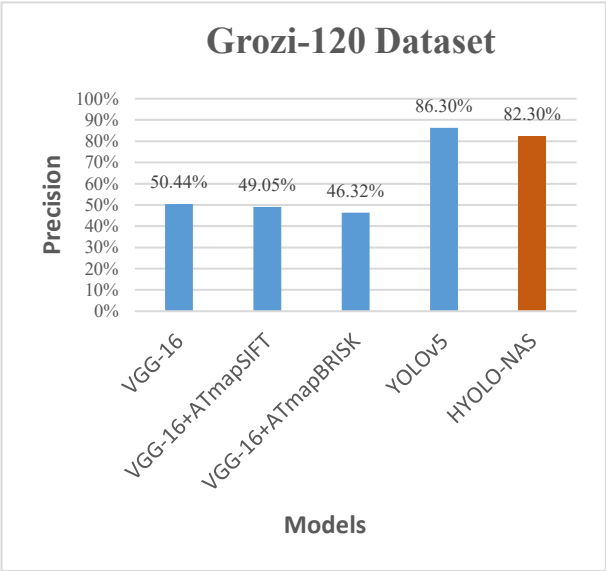

Fig. 5(a) Comparison based on Precision

Figure 5(a) shows comparative results in the graphical representation of the HYOLO-NAS model, where it performs better in mAP and F1-Score, and maintains a balance between recall and precision, making it a robust grocery detection model. The precision performance of five alternative models on the Grozi-120 dataset is depicted in Fig. 5(a). VGG-16 achieves a precision of 50.44%, VGG-16+ATmapSIFT has a precision of 49.05%, and VGG-16+ATmapBRISK has a precision of 46.32%. The YOLOv5 model has 86.30% precisio. The HYOLO-NAS model achieves precision with 82.30%. The graph illustrates that YOLOv5 and HYOLO-NAS exhibit superior precision compared to the VGG-16 versions when evaluated on the Grozi-120 dataset.

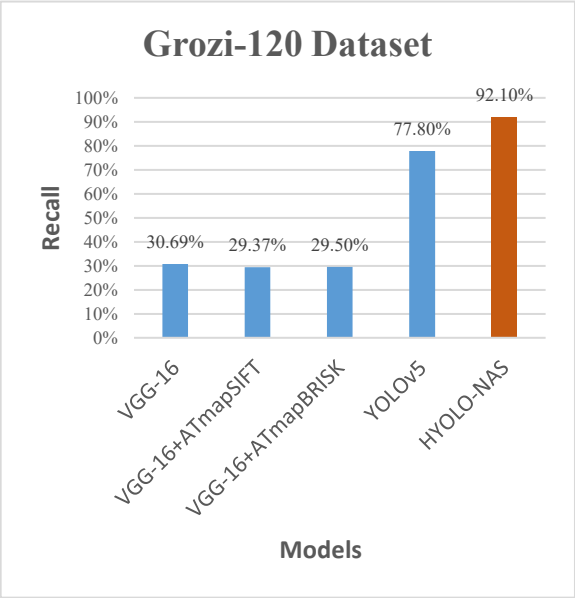

Fig. 5(b) Comparison based on Recall

The recall performance on the Grozi-120 dataset is depicted in Fig. 5(b). VGG-16 model attains a recall rate of 30.69%. The recall rate for VGG-16+ATmapSIFT is 29.37%, whereas VGG-16+ATmapBRISK has a slightly higher recall rate of 29.50%. The recall rate of YOLOv5 is 77.80%, which is noticeably higher. Our proposed HYOLO-NAS model achieves the maximum recall rate of 92.10%. The graph clearly illustrates that our suggested model, HYOLO-NAS, exhibits superior performance compared to the other models regarding recall. HYOLO-NAS identified relevant Grozi-120 instances with high recall.

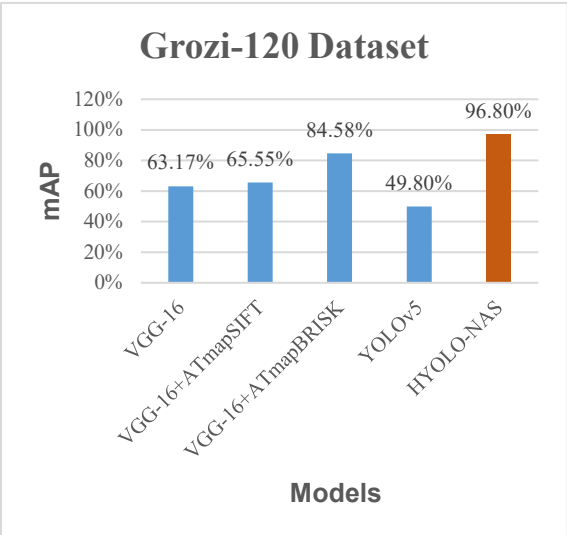

Fig. 5(c) Comparison based on mAP

Figure 5(c) illustrates the average precision (mAP) of five distinct models on the Grozi-120 dataset. VGG-16 obtains a mean average precision (mAP) of 63.17%, but VGG-16+ATmapSIFT demonstrates a minor improvement with a mAP of 65.55%. VGG-16+ATmapBRISK improves performance significantly, resulting in a mean average precision (mAP) of 84.58%. Conversely, YOLOv5 exhibits a lower mean Average Precision (mAP) of 49.80%, suggesting a less efficient performance. The examination of the results shows that our proposed model, HYOLO-NAS, is the most successful model for the Grozi-120 dataset, achieving the highest mean Average Precision (mAP) of 96.80%.

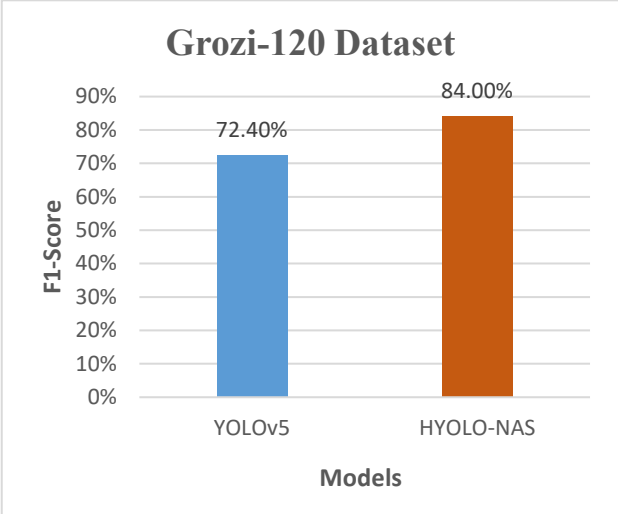

Fig. 5(d) Comparison based on F1-Score

Figure 5(d) illustrates a comparison of the F1-Scores attained by several models on the Grozi-120 dataset. The YOLOv5 model demonstrates a strong performance with an F1-score of 72.40%. However, the suggested model HYOLO-NAS demonstrates superior performance compared to YOLOv5, achieving an impressive F1-score of 84.00%. This significant 11.60% increase demonstrates the accuracy of HYOLO-NAS to identify and categorize groceries. The greater F1-score of HYOLO-NAS leads to improved overall performance.

Table 4. Experimental Results on Grozi-120 and Retail Product datasets to detect Grocery

HYOLO-NAS(Our Proposed)				
Dataset	**Precision**	**Recall**	**mAP**	**F1 Score**
Grozi-120	82.3%	92.1%	96.8%	84.0%
Retail Product	89.47%	89.50%	97.61%	89.06%

The Proposed HYOLO-NAS model exhibits strong and consistent performance across both datasets, as shown in Table 4, demonstrating its versatility and effectiveness in grocery detection scenarios. It represents that on the Grozi-120 dataset, model achieves a commendable balance between 82.3% Precision and 92.1% Recall, resulting in F1-Score of 84.0% with high mAP of 96.8%. On the Retail Product dataset, the model achieves even higher precision (89.47%) and

comparable Recall (89.50%), leading to an outstanding F1-Score of 89.06%. The mAP of 97.61% highlights the model's capability.

Fig. 6: Retail Product dataset used to train proposed Hypertuned YOLO-NAS Model

7(a) Visual detection 1

7(b) Visual detection 2

Fig. 7: Results of Hypertuned YOLO-NAS on Retail Product dataset

Retail Product datasets are graphically illustrated in Fig. 6 and some visual detection samples are presented in Fig. 7 to accurately and consistently detect grocery products.

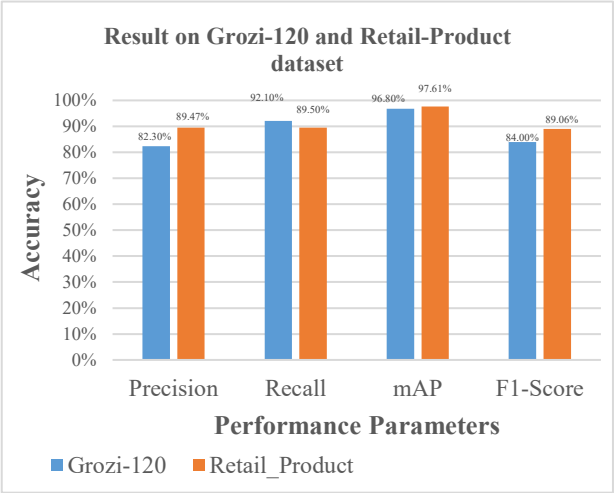

Fig. 8: Comparative Analysis of Grozi-120 and Retail Product using HYOLO-NAS

Figure 8 shows HYOLO-NAS's performance on both Grozi-120 and Retail Product datasets. The model demonstrates a precision of 82.30% on the Grozi-120 dataset and 89.47% on the Retail Product dataset, indicating a superior ability to identify meaningful objects within the Retail Product dataset accurately. This suggests that the model is more efficient in minimizing false positives, specifically in retail products. With a slightly higher recall on the Grozi-120 dataset, the model's recall scores of 92.10% for Grozi-120 and 89.50% for Retail Product demonstrate its strong ability to detect the most relevant objects. The model accurately ranks observed objects proficiently, with a mean Average Precision (mAP) of 96.80% on Grozi-120 and 97.61% on Retail Product datasets. The F1-Scores of 84.00% on Grozi-120 and 89.06% on Retail Product demonstrate a harmonious balance between precision and recall. The higher F1-score on Retail Products indicates that HYOLO-NAS achieves a favorable balance, resulting in efficient detection and classification. The outstanding performance metrics in all parameters indicate that HYOLO-NAS is a robust and versatile model, making it highly successful on both datasets.

7. Conclusion

YOLO-NAS is quantization-friendly, a new object detection model that is faster than previous YOLO models. In the proposed work, the HYOLO-NAS model performed a detection on grocery images that can help the visually impaired by converting text to audio in their real-time scenarios. To improve accuracy, the model's parameters such that epochs, learning rate with linear warm up with Cosine Annealing function, and L2 regularization weight decay of 0.0001 into the ADAM optimizer are optimized, which efficiently constructed HYOLO-NAS for grocery products. Results show that the model has increased Recall by 14.3%, mAP by 12.22%, and F1 score by 11.6% on Grozi-120. Meanwhile, it also has promising results on the Retail

Product dataset, having a Precision of 89.47%, Recall of 89.5%, mAP 97.61% and F1-Score of 89.06%. This demonstrates optimized Hypertuned YOLO-NAS models' potential for practical use cases where other object detection models might fall short in real-time when detecting objects.

References

1) Wei, Yuchen, Son Tran, Shuxiang Xu, Byeong Kang, and Matthew Springer, "Deep learning for retail product recognition: Challenges and techniques," *Computational Intelligence and Neuroscience 2020* Article ID 8875910 (2020).https://doi.org/10.1155/2020/8875910.
2) D. Ravi Kumar, Hiren Kumar Thakkar, Suresh Merugu, Vinit Kumar Gunjan, "Object Detection System for Visually Impaired Persons Using Smartphone," *In book: ICDSMLA* pp. 1631-1642 (2020). doi:10.1007/978-981-16-3690-5-154 .
3) Behzad Mirzae, Hossein Nezamabadipou, Reza Derakhshan, Amir Raoo, "Small Object Detection and Tracking: A Comprehensive Review," Advances in Deep Learning Based Sensing, Imaging, and Video Processing, Sensors 2023, 23(15), 6887 (2023). https://doi.org/10.3390/s23156887.
4) Terven, Juan, and Diana Cordova Esparza, "A comprehensive review of YOLO: From YOLOv1 to YOLOv8 and beyond," arXiv preprint arXiv:2304.00501 (2023).
5) Terven, Juan, Diana Margarita C´ordova Esparza, and Julio Alejandro Romero Gonz´alez, "A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS," Machine Learning and Knowledge Extraction 5, no. 4, 1680-1716, (2023). doi:10.3390/make5040083.
6) "YOLO-NAS by Deci Achieves SOTA Performance on Object Detection Using Neural Architecture Search," (2023). https://deci.ai/blog/yolo-nas-object-detection-foundation-model.
7) White, Colin, Mahmoud Safari, Rhea Sukthanker, Binxin Ru, Thomas Elsken, Arber Zela, Debadeepta Dey, and Frank Hutter, "Neural architecture search: Insights from 1000 papers," arXiv preprint arXiv:2301.08727 (2023). https://doi.org/10.48550/arXiv.2301.08727
8) Baymurzina, Dilyara, Eugene Golikov, and Mikhail Burtsev, "A review of neural architecture search," Neurocomputing 474 82-93 (2022). https://doi.org/10.1016/j.neucom.2021.12.014.
9) Reyad, Mohamed, Amany M. Sarhan, and M. Arafa, "A modified Adam algorithm for deep neural network optimization," Neural Computing and Applications Open access, **volume 35**, pages 17095–17112 (2023).
10) Mengzi Hu, Ziyang Li, Jiong Yu, "Efficient Lightweight YOLO: Improving Small Object Detection in YOLO for Aerial Images," Sensors, 23(14) 6423 (2023) https://doi.org/10.3390/s23146423.
11) Aduen Benjumea, Izzeddin Teeti, Fabio Cuzzolin, Andrew Bradley, "YOLO-Z: Improving small object detection in YOLOv5 for autonomous vehicles," Computer Vision and Pattern Recognition, (2023). https://doi.org/10.48550/arXiv.2112.11798.
12) Hongxia Yu, Lijun Yun, Zaiqing Chen, Feiyan Cheng, and Chunjie Zhang, "A Small Object Detection Algorithm Based on Modulated Deformable Convolution and Large Kernel Convolution," Computational Intelligence and Neuroscience, Article ID 2506274, (2023). https://doi.org/10.1155/2023/2506274.
13) Shu Jun Ji, Qing Hua Ling, Fei Han, "An improved algorithm for small object detection based on YOLO v4 and multi-scale contextual information," Computers and Electrical Engineering, **Volume 105** 108490,(2023). https://doi.org/10.1016/j.compeleceng.2022.108490.
14) Prabu Selvam, Joseph Abraham Sundar Koilraj, "A Deep Learning Framework for Grocery Product Detection and Recognition," Research Square, 2022. DOI: https://doi.org/10.21203/rs.3.rs-1431986/v1.
15) Yuchen Wei, Son Tran, Shuxiang Xu, Byeong Kang, and Matthew, "Deep Learning for Retail Product Recognition: Challenges and Techniques," Computational Intelligence and Neuroscience, **Volume 2020** Article ID 8875910 (2020). https://doi.org/10.1155/2020/8875910.
16) Gothai, Surbhi Bhatia, Aliaa M. Alabdali, Dilip Kumar Sharma, Bhavana Raj Kondamudi and Pankaj Dadheec, "Design Features of Grocery Product Recognition Using Deep Learning," Intelligent Automation and Soft Computing 34(2):1231-1246 (2022). doi:10.32604/iasc.2022.026264.
17) Zhiwei Lin, "HS-YOLO: Small Object Detection for Power Operation Scenarios," Applied Science 2023, 13(19) 11114 (2023). https://doi.org/10.3390/app131911114.
18) Tanvir Ahmad, Yinglong Ma, Muhammad Yahya, "Object Detection through Modified YOLO Neural Network," Scientific Programming (2020). doi:10.1155/2020/8403262 License Cc By 4.0.
19) zhuang zhuang wang, kai xie, xin yu zhang, hua quan chen, chang wen, and jian biaohe, "Small Object Detection Based on YOLO and Dense Block via Image Super Resolution," IEEE Access, License CC BY 4.0 PP(99): 1-1 (2021). doi:10.1109/ACCESS.2021.3072211.
20) Tao Sun, Haonan Chen, Haiying Liu, Haitong Lou1, Xuehu Duan, "HPS-YOLOv7: A High Precision Small Object Detection Algorithm," research square, (2023). https://doi.org/10.21203/rs.3.rs-2813484/v1.
21) Pengzhen Ren, Yun Xiao, Xiaojun Chang, Poyao

Huang, Zhihui Li, Xiaojiang Chen, Xin Wang, "A Comprehensive Survey of Neural Architecture Search: Challenges and Solutions," ACM Computing Surveys 2021, **Volume 54** Issue 4 Article No. 76 pp.1–34 (2021). https://doi.org/10.1145/3447582.

22) Arjun Saud, Subarna Shakya, "Analysis of L2 Regularization Hyper Parameter for Stock Price Prediction," *Journal of Institute of Science and Technology* 26(1):83-88 License CC BY-SA 4.0 (2021). doi:10.3126/jist.v26i1.37830.
23) Sana Ben Hamida, Hichem Mrabet, Faten Chaieb, Abderrazek Jemai, "Assessment of data augmentation, dropout with L2 Regularization and differential privacy against membership inference attacks," Multimedia Tools and Applications, (2023). doi:10.1007/s11042-023-17394-3.
24) Ahmed Husham Al Badri, Nor Azman Ismail, Khamael Al Dulaimi, Ghalib Ahmed Salman, Md Sah Hj Salam, "Adaptive Non-Maximum Suppression for improving performance of Rumex detection," Expert Systems with Applications, (2023). https://doi.org/10.1016/j.eswa.2023.119634.
25) Wang Xin Li, Xin Lyu, Tao Zeng, Jiale Chen, Shangjing Chen, "Multi Attribute NMS: An Enhanced Non-Maximum Suppression Algorithm for Pedestrian Detection in Crowded Scenes," Deep Learning in Object Detection and Tracking, Appl. Sci.2023 13(14) 8073 (2023). https://doi.org/10.3390/app13148073.
26) Rasmus Rothe, Matthieu Guillaumin, Luc Van Gool, "Non-Maximum Suppression for Object Detection by Passing Messages between Windows," Conference: Asian Conference on Computer Vision (2015). doi:10.1007/978-3-319- 16865-4-19.
27) Iza Sazanita Isa, Mohamed Syazwan Asyraf Rosli, Umi Kalsom Yusof, Mohd Ikmal Fitri Maruzuki, And Siti Noraini Sulaiman, "Optimizing the Hyperparameter Tuning of YOLOv5 for Underwater Detection," IEEE Access, (2022). doi: 10.1109/ACCESS.2022.3174583.
28) R. Anita Jasmine, P. Arockia Jansi Rani, J.Ashley Dhas, "Hyper Parameters Optimization for Effective Brain Tumor Segmentation with YOLO Deep Learning," J*ournal of Pharmaceutical Negative Results* (2022). doi:10.47750/pnr.2022.13.S06.292
29) Kehao Du, Alexander Bobkov, "An Overview of Object Detection and Tracking Algorithms," Presented at the 15th International Conference Intelligent Systems (INTELS'22), Moscow, Eng. Proc.2023,33(1), 22; (2023). https://doi.org/10.3390/engproc2023033022
30) Yadav Satya Prakash, Muskan Jindal, Preeti Rani, Victor Hugo C. de Albu querque, Caiodos Santos Nascimento, and Manoj Kumar, "An improved deep learning based optimal object detection system from images," Multimedia Tools and Applications 1-28 (2023). https://doi.org/10.1007/s11042-023-16736-5
31) Ankit Sinha, Soham Banerjee, Pratik Chattopadhyay, "An Improved Deep Learning Approach For Product Recognition on Racks in Retail Stores," 2022, Computer Vision and Pattern Recognition, https://doi.org/10.48550/arXiv.2202.13081.
32) NhatDuy Nguyen, Tien Do, Thanh Duc Ngo, and DuyDinh Le1, "An Evaluation of Deep Learning Methods for Small Object Detection," *Journal of Electrical and Computer Engineering*, **Volume 2020** Article ID 3189691 (2020). https://doi.org/10.1155/2020/3189691.
33) YOLO-NAS by Deci Achieves, "SOTA Performance on Object Detection Using Neural Architecture Search," May 3 (2023). https://deci.ai/blog/yolo-nas-object-detection-foundation-model.
34) Mohand Lagha, Kheireddine Choutri, Souham Meshoul, Mohamed Batouche, Farah Bouzidi, Wided Charef, "Fire Detection and Geo-Localization Using UAV's Aerial Images and Yolo-Based Models," Applied Science 13(20) 11548 (2023). https://doi.org/10.3390/app132011548.
35) Chenhao He, "Investigating YOLO Models Towards Outdoor Obstacle Detection For Visually Impaired People," Research Square, (December 2023). doi:10.21203/rs.3.rs-3733857/v1.
36) Zaipeng Xie, Zhaobin Li, Yida Zhang, Jianan Zhang, Fangming Liu, Wei Chen, "A MultiSensory Guidance System for the Visually Impaired Using YOLO and ORB-SLAM," License CC BY 4.0 13(7):343 (July 2022). doi:10.3390/info13070343.
37) Fitria Dewi, "RETAIL PRODUCT Dataset," (2023).Roboflow. https://universe.roboflow.com/fitria-dewi-c1vvv/retail-product-xeisr.
38) etVisPublic, "Grozi120 Dataset," Roboflow. (2022). https://universe.roboflow.com/retvispublic-dutr5/grozi120.
39) Alexey Bochkovskiy, Chien Yao Wang, "YOLOv4: Optimal Speed and Accuracy of Object Detection," Computer Vision and Pattern Recognition, (2020). https://doi.org/10.48550/arXiv.2004.10934.
40) Deci, "YOLO-NAS: A New SOTA Model For Object Detection," (May 03, 2023). https://docs.ultralytics.com/models/yolo-nas/.
41) Karlinsky, L., Shtok, J., Tzur, Y., and Tzadok, A, "Fine-grained recognition of thousands of object categories with single example training," In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4113- 4122 (2017).
42) Geng, Weidong, Feilin Han, Jiangke Lin, Liuyi Zhu, Jieming Bai, Suzhen Wang, Lin He, Qiang Xiao, and Zhangjiong Lai, "Fine-grained grocery product recognition by one-shot learning," In Proceedings of the 26th ACM international conference on Multimedia, pp. 1706-1714 (2018).
43) Santra, Bikash, Udita Ghosh, and Dipti Prasad Mukherjee, "Graph-based modelling of superpixels

for automatic identification of empty shelves in supermarkets," Pattern Recognition Volume 127 Issue C (Jul 2022). https://doi.org/10.1016/j.patcog.2022.108627.

44) Selvam, Prabu, and Joseph Abraham Sundar Koilraj. A deep learning framework for grocery product detection and recognition. Food Analytical Methods 15, no. 12 (2022), 3498-3522.
45) Yadav, S.P., Zaidi, S., Mishra, A., "Survey on Machine Learning in Speech Emotion Recognition and Vision Systems Using a Recurrent Neural Network (RNN)," Arch Computat Methods Eng 29, 1753–1770 (2022). https://doi.org/10.1007/s11831-021-09647-x
46) Kakde, A., Arora, N., & Sharma, D. (2018). Novel Approach towards Optimal Classification using Multilayer Perceptron. International Journal of Research in Engineering, IT and Social Sciences, 810, 29-38.
47) Vandana, K. Kumar Yogi and S. P. Yadav, "Surveillance to Detect and Classifying of Chicken Poultry Diseases from Fecal Images Using CNN," 2023 6th International Conference on Contemporary Computing and Informatics (IC3I), Gautam Buddha Nagar, India, 2023, pp. 939-943, doi: 10.1109/IC3I59117.2023.10397876.