Review

Exploring the Depths of Visual Understanding: A Comprehensive Review on Real-Time Object of Interest Detection Techniques

EBERE UZOKA CHIDI *, COLLINS N. UDANOR *, EDWARD ANOLIEFO *

Posted Date: 9 February 2024

doi: 10.20944/preprints202402.0583.v1

Keywords: Object detection, object of interest detection, object of instance detection, deep learning.

Review

Exploring the Depths of Visual Understanding: A Comprehensive Review on Real-Time Object of Interest Detection Techniques

Ebere Uzoka [1], Collins Udanor [2] and Edward Anoliefo [3]

[1] University of Nigeria, 410105 Enugu, Nigeria
[2] University of Nigeria, 410105 Enugu, Nigeria; collins.udanor@unn.edu.ng
[3] University of Nigeria, 410105 Enugu, Nigeria; edward.anoliefo@unn.edu.ng
* Correspondence: uzoka.ebere116@unn.edu.ng

Abstract: Object detection is complex and involved diverse requirements for different applications. In critical application such as visual impaired navigation guide and real-time surveillance for instance, identifying a particular object of instance is required and this involved a complex approach for realization. Literatures were reviewed, starting with the application of deep learning techniques for object detection; their gaps were identified and addressed using real-time object detection models literature review. From the review gaps were also detected and addressed using literature review on occlusion detection techniques. Object of instance detection in clustered scene with similar object of interest was identified as an open gap not addressed in the existing literature; even through it is vital for many applications. This research recommended an Occlusion Based Object of Instance Detection (OBOID) technique which used the spatial information to identify instant object of interest in clustered scene with many similarly object of interest. Limitation of the recommended OBOID is that it requires only system where position and distance of object is necessary to inform other decision.

Keywords: Object detection; object of interest detection; object of instance detection; deep learning.

1. Introduction

Computer Vision Technology (CVT) is an Artificial intelligence (AI) based systems that allow computers to see, localize the position of objects and recognize the contents of digital images or video [1]. Today this CVT are popular in applications such as image classification [2], real-time object detection [3], pose estimation [4], segmentation [5], video classification [6], object tracking [7], and super-resolution [8]. The main processes to achieve image classification involve data collection [9], data process [10], feature extraction [11], and classifier [12]. Image classification is a process of assigning a label to the class of an image [2]. It identifies the existing pixels in an image and then uses the information to determine the class of the image through a trained classifier. These classifiers are specialized Convolutional Neural Network (CNN), algorithms such as Alex.Net [13], ResNet [14], Mobile.Net [15], and DenseNet, [16] trained with the image features to generate the classification models. While these deep learning models are successful in solving image classification problems, and are currently applied or object detection; one of the biggest challenges is localizing the image classified through real-time object detection and tracking [17].

Object detection is a computer vision technique used for the identification and localization of an object in a video frame [18]. It is classified into single-stage-detectors such as short single multi-box detectors [18], You-Can-Only-Look-Once (YOLO) series [19,20] and then 2-stage-detectors such as Recurrent–CNN (R-CNN) [21], Fast-R-CNN [22], Mask-CNN [23] and Faster-R-CNN [24]. According to [24] 1-stage object detection models are more reliable for real-time object detection tasks because of their speed of detection and classification when compared with the 2-stage counterparts.

 doi:10.20944/preprints202402.0583.v1

According to [26], while object detection models have greatly improved through innovations such as 3D models, robust feature detection and real-time image matching [27], one of the biggest challenges remained how to detect objects of interest in clustered scenes [28], detection of objects with dynamic behavior [29], detection of actionable objects [27], real-time object detection [29] and Occlusion [30]. To address these problems, You Can Only Look Once (YOLOV) algorithms have dominated models, improved with specialized segmentation techniques to solve such as multiple object tracking algorithms [35], region of interest detection algorithms [36], adaptive filtering techniques [37,38], data augmentation [39,40] context-aware approach [41], grid virtual division [42], ClusterNet [43], Few-Shot Object Detection (FSOD) [44], and deep learning [45,46]. This paper will review the application with YOLOV for real-time object detection using these techniques and identify current challenges and recommendations for improvement.

The Organization and Contribution of This Paper

The paper begin with an overview of deep learning techniques, identify popular pre-trained model developed with CNN as the foundation. The impact of these models for object detection and classification was discussed using literature review and research gap identified. To bridge the gap, more literatures on real-time object detection and tracking techniques specifically YOLOV models was presented, and then a new gap was also identified. In the same vein, Occlusion detection models were reviewed to address the YOLOV gaps, and then issues of detecting objects of instance (occlusion problem when there are multiple object of interest in a scene) were identified as unsolved gaps currently exiting. The new techniques was recommended as solution to solve the problem.

2. Overview of Deep Learning Techniques

Deep Learning (DL) is a subset of machine learning that employs multiple layers of neural networks for the modeling of intricate patterns and relationships within a dataset [47,48]. Popular DL techniques are CNN and Recurrent Neural Network (RNN). CNN according to [48] specializes in solving image classification problems, while RNN are powerful DL tool specialized in solving sequential data such as speech recognition and natural language processing [49]. Today CNN has become a cornerstone in image classification problems with several factors such as spatial hierarchy and locality, parameter sharing, local connectivity of feature maps, and pooling translation invariance [50]. Figure 1 presents the architecture of CNN, with components such as input layers, convolutional layers, fully connected layers, output layers, pooling layers, feature maps, and activation functions [51,52].

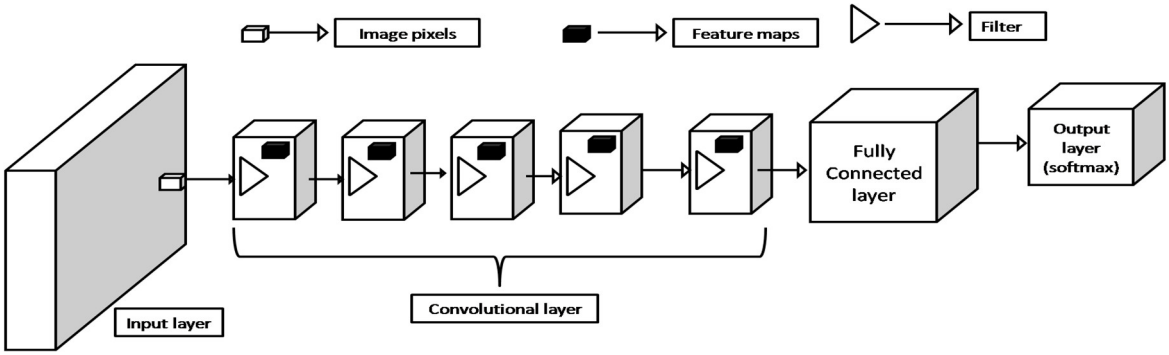

Figure 1. Block diagram of CNN.

Figure 1 presents the block diagram of a CNN with an input layer responsible for input image dimension considering height, weight and color channel. The image pixels are identified by a filter that performs a convolutional scan at the image receptive fields to extract feature maps and then pool using pooling techniques such as maximum pooling or average pooling [53]. The CNN in this case has five convolutional layers each formulated from the convolutional scan of the previous layer. At the end of the convolutional process, the feature vectors are flattened and fed to the fully connected

 doi:10.20944/preprints202402.0583.v1

layer where batch normalization and optimization algorithms are used to train the neural network in the FCL. The softmax function is the activation function that transforms the features into a probability score in the output layer. While CNN has recorded great success for image classification problems, it has struggled to replicate the same success on small datasets due to issues such as over-fitting, slow convergence, poor generalized model, Shit invariance, and bias [54,55] and hence presents the need for pertained models. These are pre-trained as a heterogeneous CNN architecture trained with a large corpus of datasets to form the building block when developing new models and hence address these aforementioned traditional CNN challenges [56]. Popular CNN-based pre-trained models are Alex.Net, ResNet, Mobile.Net, and DenseNet [57], while many other pre-trained models are GoogleNet [58], Inception-V3 [59], VGG16 [60] and VGG19 [61], Inception-ResNet [62], DarkNet [63], Xception [64], ShuffleNet [65], and SqueezeNet [66].

2.1. Popular CNN Model and Their Architectures

This section will discuss popular CNN-based pre-trained models such as Alex.Net, ResNet, Mobile.Net, and DenseNet, identifying the major components that constitute them and discussing their functions.

1. **Mobile-Net**

Mobile-Net is another popular pre-trained model specially designed for image classification problems on devices with limited computational resources like mobile applications [53,67]. It is a lightweight model with multiple layers with the ability to adapt when fine-tuned as a transfer learning algorithm. Figure 2 presents the architecture of a Mobile-Net.

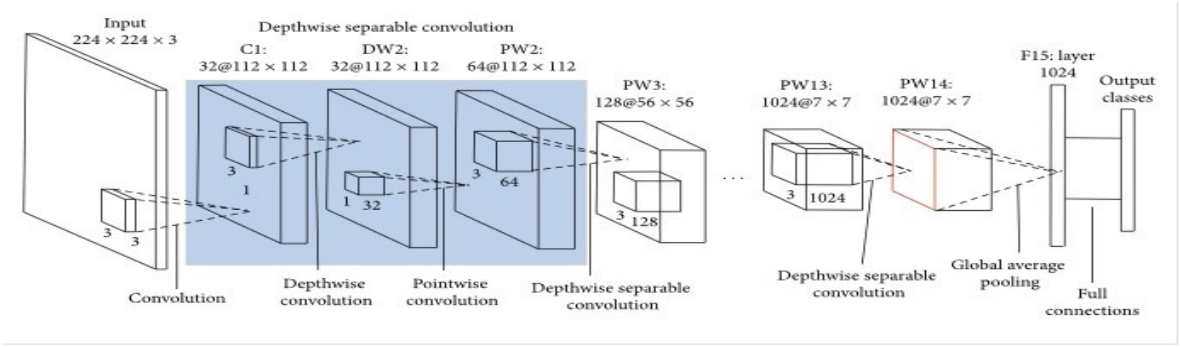

Figure 2. The architecture of Mobile-Nets [53].

Figure 2 illustrates a Mobile-Net and its layers. The input layer dimensions the image size into 224*224*3 (height, weight and color channel), then convolution is applied on the image strides to extract the feature maps into the depth-wise layer and point-wise layer which doubles the number of channels for the global average pooling for better output and full connections [68]. The depth-wise layer uses a single filter per input channel to minimize complexity, while the point-wise uses its convolutional process to fusion feature maps across channels and global average pooling applied to extract the features until the final layer, with reduced image size and increased color channel [13]. Overall in the image input, 224*224*3 was transformed through the application of the MobileNet components to 7*7*1024, with the color channel. This increased transformation in color channel plays a significant role in the Mobile-Net classification efficiency, reduces parameters and ensures model compactness during deployment [69].

2. **Alex.Net**

Alex.Net is one of the popular object detection algorithms developed with multiple layers inspired by CNN architecture as a pre-trained model for object detection and recognition. Since its innovation by Alex et al. in 2012 using the ImageNet dataset with 1000 classes, it has gained increased attention, particularly for image classification problems [70]. The Alex.Net in Figure 3 is made of the input layer, five convolutional layers, three fully connected layers and the output layer [71].

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 9 February 2024 doi:10.20944/preprints202402.0583.v1

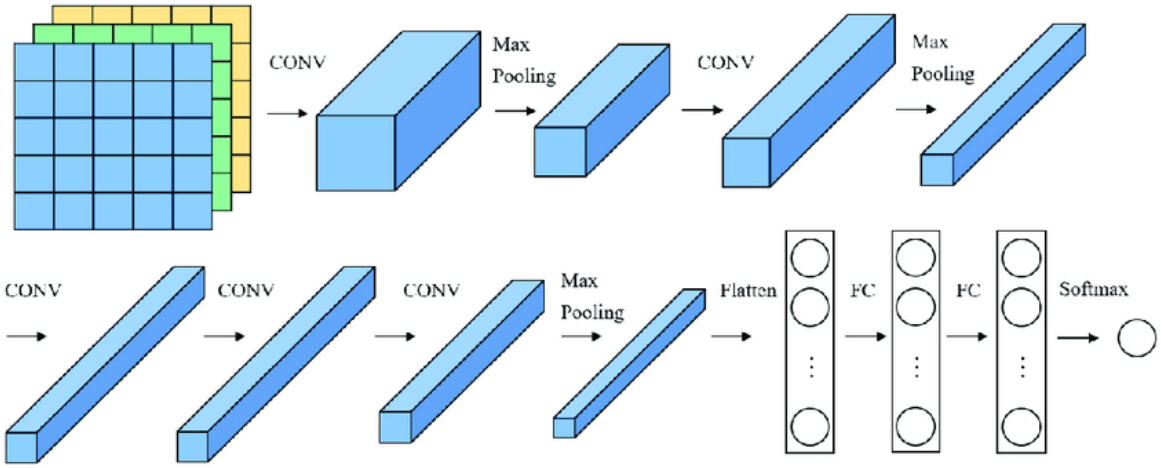

Figure 3. Alex.Net architecture [70].

The input layer of the Alex.Net in Figure 3 is the first layer which is the access to import data to the network. The imported data is dimensioned in the layer and then the features are scanned using a convolutional filter and then maximum pooling is applied to extract the identified features maps and formulate the convolutional layer (CONV). The process continues until the final convolutional layer where the features are flattened and then fed to the Fully Connected Layer (FC). The first FC is made of 4096 neurons which combine the high-level features map flatten and form a deeper representation of the image, this process is further refined in the second FC to form a more complex representation of the image, before the final FC which has 1000 neurons corresponding to the 1000 classes in the training ImageNet, each presenting a class of the output triggered by softmax activation function [72].

3. **DenseNet**

DenseNet is another deep-learning model designed for image classification problems. It is a densely connected convolutional neural network that serves different purposes in computer vision applications [73]. This dense connectivity of neurons ensures that feature maps can be re-used, thereby solving the vanishing gradient problem, improving the propagation of features and reducing the number of training parameters [74]. Figure 4 presents a block diagram of DenseNet.

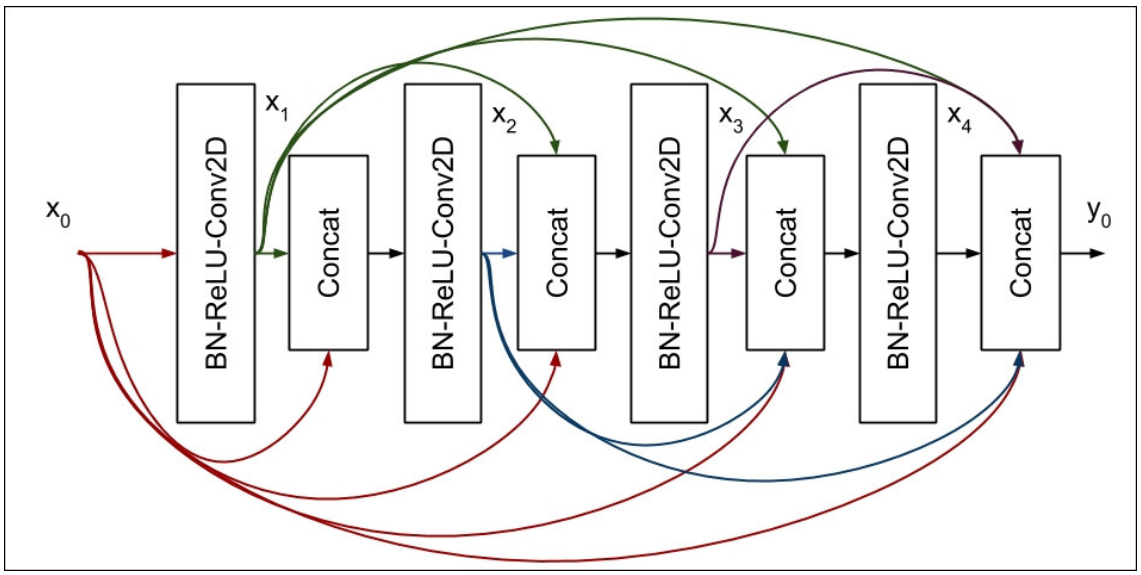

Figure 4. DenseNet Architecture.

Figure 4 showcased the components of DenseNet which are made of dense blocks, a bottleneck layer and a transition layer. The dense block is made of Batch Normalization Rectified Linear Unit

 doi:10.20944/preprints202402.0583.v1

Convolutional 2D (BN-ReLU-conv2D) and Concatenation (Concat) [73]. The BN-ReLU-Connv2D helps in stabilizing and accelerating the process of feature extraction, while Concat merges the extracted information from different pathways within the network dense network structured in this case of five layers ($X_o \ldots\ldots\ldots\ldots\ldots\ldots X_5$) [73,74]. The bottleneck layer manages the number of channels using a compression factor to control computational efficiency while the transformation layer applies the pooling process to minimize the spatial dimension of the image size [75].

4. **Residual Network (ResNet)**

Residual Network (ResNet) introduced in 2016 by Kaiming et al. is another deep learning model designed with the primary aim of addressing training complexities with deep neural networks using residual block [76]. Over the years, ResNet has been applied for real-time classification of objects in various applications [77]. Figure 5 presents the architecture of ResNet with its major components which are the input layer, residual block, fully connected layer, pooling, softmax function and output.

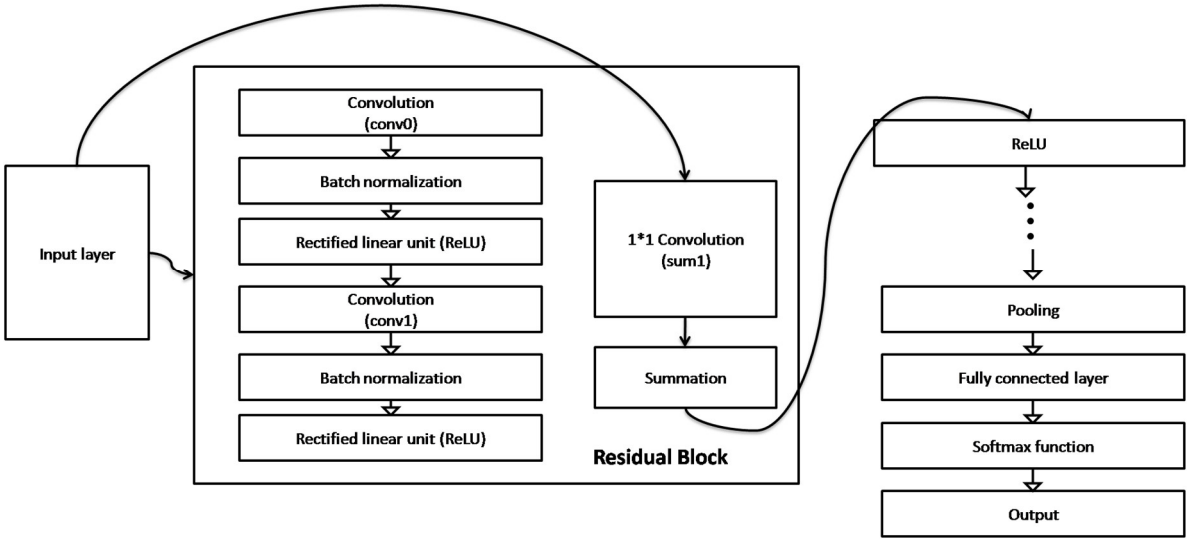

Figure 5. Block diagram of ResNet.

Figure 5 showcased the architecture of a ResNet. It contains the input layer which takes the input data, initial convolution and pooling used on the input data to extract features, residual stages are divided into stages, and each stage consists of several residual blocks that help in activating functions, global average pooling used to reduce the spatial dimensions to a 1x1 size and fully connected layer used for the final classification [76]. The convolutional layers (conv0, conv1), batch normalization and ReLu collectively extract features of the input images and sum at the output of the residual block for pooling and transformation into the fully connected layer (FC), where it is trained and output using softmax function to predict the classification results [77].

2.2. Literature on Image Classification Problem Using Pre-Trained Models

This section will review relevant literature that applied pre-trained models such as Alex.Net, ResNet, Mobile.Net, and DenseNet for image classification problems. The literature will be reported systematically and at the end of every section, a summary of the review, findings and research will be presented.

1. **Relevant Literatures on image classification with Alex.Net**

Aapplied fine-tuned Alex.net with a CNN model trained with the Cifar-10 version-2019 dataset for object detection and recorded 98% classification accuracy [78]. Also trained CNN with four brain tumor datasets which are Figshare, Brain MRI Kaggle, Medical MRI datasets and BraTS 2019 datasets to generate a model used to fine-tune Alex.Net as a 3D model for the classification of brain tumours [79]. The average performance for accuracy is 99%, mean Average Precision (mAP) reported 99%, sensitivity 87% and detection time of 1300ms. While these models recorded very high accuracy for

 doi:10.20944/preprints202402.0583.v1

object detection and classification, argued that the delay in classification makes Alex.net not the most suitable for real-time object detection problems [25]. Another study [70] applied Alex.net for the classification of eye behavior using the Zhejiang University (ZJU) and Closed Eyes in the Wild (CEW) dataset. The average results recorded for both datasets, considering parameters such as accuracy, sensitivity, specificity, and precision, are 95% respectively which is good, however [71] argued that during the training of the Alex.Net, traditional training algorithms such as gradient descent has potential for over-fitting, especially with a small dataset. This was addressed using an improved Adam optimization technique achieved with corrosion expansion and Gaussian filtering [72]. In addition, the quality of data collection was improved using features of weight sharing, local connection, and learning image representation. This was applied to Alex.net for the classification of oil spillage. The results when tested reported 99.76% for recall and 98.91% for accuracy. While most of these models lack practical validation despite their success, applied Field Programmable Gate Arrays (FPGA) hardware to validate the Alex.net model for real-time object detection [80]. However, [19–21] argued that for real-time classification tasks, the speed of object detection is vital, and the performance of Alex.Net with 0.78seconds [81] and 1.24seconds [82], suggested that while it is perfect for the classification problem, it is not the most suitable to address real-time object detection problems. In the context of a Navigation guidance system for the blind, the speed of object detection is very important to inform immediate decisions to avoid accidents, however when there is a delay, this impacts the system responsiveness and also the user's ability to react to obstacle avoidance along the path of navigation, thus leading to collision and then affects user confidently and safety. Therefore it is necessary to adopt a model which not only detects and recognize objects with high accuracy, but also in real-time. Table 1 was used to summarize the systematic literature review on Alex.net for object detection problems.

Table 1. Non- exhaustive summary of Alex-Net review on object detection.

Authors	Primary objectives	Methods	Findings	Research gap
[78]	Object detection	Alex.net was fined with a CNN trained on Cifar-10 for object detection	The results recorded 98% classification accuracy	Occlusion was not addressed
[79]	Fine tune Alex.et as a 3D model for the classification of brain tumor.	Trained CNN with four brain tumor datasets which are Figshare, Brain MRI Kaggle, Medical MRI datasets and BraTS 2019 datasets	The average performance for accuracy is 99%, mAP reported 99%, sensitivity 87% and detection time of 1300ms.	Sensitivity can still be improved
[70]	Classification of eye behaviour	Applied Alex.net using the Zhejiang University (ZJU) and Closed Eyes in the Wild (CEW) dataset.	The average results recorded for both datasets, considering parameters such as accuracy, sensitivity, specificity, precision, are 95% respectively which is good	Not reliable for real-time classification due to delay
[75]	Optimization of Alex.Net against overfitting	Guassian filter and Adam optimization was used to improve Alex.Net training and applied for the detection of oil spillage	The results when tested reported 99.76% for recall and 98.91% or accuracy.	Suffer delay during real-time application
[80]	Real-time classification	Applied Field Programmable Gate Arrays (FPGA) hardware to validate Alex.net model	Demonstrated image classification ability	Suffer delay

2. Relevant Literatures on object detection with Mobile-Net

Mobile-Net over the years has been applied to solve object detection problems [53]. Single-Shot Detector (SSD) and OpenCV library were applied in [15] to optimize the object detection capabilities

 doi:10.20944/preprints202402.0583.v1

of Mobile-Net to generate a Mobile-Net-SSD. The SSD operated as a multiscale object detector of feature maps from different layers of the Mobile-Net and then applied a bounding box to predict the position of the object [67]. The Mobile-Net-SSD was trained on MS Common Objects in Context (COCO). The results reported 0.92 for accuracy, recall of 0.81 and mAP of 0.85, which is good but leaves room for improvement on both performance metrics. Similarly [68] applied SSD-Mobile-Net-2 for object detection and obstacle avoidance in Autonomous Driving Assistance System (ADAS). The results when tested on five different objects reported an average accuracy of 97.8%, while [69] who also applied the SSD for multi-scale feature map detection and bounding box prediction in Mobile-Net for real-time object detection reported an average accuracy of 89.53% which is ok, but leaves room for improvement. From the literature review, it was observed that while Mobile-Net is carefully designed for lightweight applications systems with less computing resources, during object detection, its overall accuracy leaves room for improvement when compared with another deep learning model such as YOLOV [83]. More so [84] revealed after comparing Mobile-Net with YoloV-5 on three different Google Cloud Platforms (Nvidia Telse, 1160 and Jatson Nano) respectively showcased that Mobile-Net has less object detection speed than YOLOV-5. What this means concerning the guide navigation application system is that, with improvement needed in speed and accuracy despite its success, it may not be the most suitable model of adoption for real-time classification problems. This is because speed is a very vital factor that informs other processes such as the audio feedback mechanism, hence when the speed and accuracy of classification is not optimal, its reliability for navigation by the blind will be affected. Table 2 presents a summary of the literature reviewed on Mobile-Net for object detection and classification.

Table 2. Non- exhaustive summary for Mobile-Net.

Authors	Primary objectives	Methods	Findings	Research gap
[15,67]	Object detection	The Mobile-Net-SSD was trained on MS Common Objects in Context (COCO)	The results reported 0.92 for accuracy, recall of 0.81 and mAP of 0.85,	Overall results need improveme nt
[68]	Object detection and obstacle avoidance	SSD-Mobile-Net-2	The results when tested on five different objects reported an average accuracy of 97.8%.	It is not interactive
[69]	Multi-scale feature map detection and bounding box prediction in Mobile-Net for real-time object detection	SDD, Mobile-Net	The results reported an average accuracy of 89.53%.	Occlusion was not addressed
[84]	Comparative study on Mobile-Net and YOLOV for object detection	Mobile-Net, YOLOV-5, Nvidia Telse, 1160 and Jatson Nano	Mobile-Net has less object detection speed than YOLOV-5	It is not interactive

3. Relevant Literatures on object detection with DenseNet

Applied two DenseNet models for object detection using Pascal VOC2007 dataset [85]. First is the F-RCNN [22,23] based DenseNet where the dense network extracts multiscale features from the images, and then the F-RCNN acts as the predictor using object regions using a bounding box. Also, SSD [53] was applied as a multiscale object detector and bounding box predictor for another DenseNet. The two models when comparative analyzed reported mAP of 5.63 with F-RCNN-based DenseNet and 3.86 with SSD-based DenseNet. Also when tested on video data, an improved mAP of 0.9854 was reported in [86] when SDD-based DenseNet was applied for object classification. The result for precision also reported 98.5%, while recall recorded 97.0%, which suggested the SDD-DenseNet performs better for video image classification than a standard image dataset. In another

 doi:10.20944/preprints202402.0583.v1

study, applied Region Proposed Network (RPN) to identify potential region of interest for processing by the CNN or prediction [73], after the DenseNet has extracted the features maps from the input image. The RPN-based DenseNet was trained for roadside object detection using PASCAL VOC and MS COCO datasets. The result reported for PASCAL VOC reported 80.30% mAP and MS COCO dataset reported mAP of 55.0%, while [74] used Deep Pyramidal Residual Networks (DPRN) and DenseNet for image recognition. The DPRN was applied to facilitate the training performance of the DenseNet, through a parallel convolutional feature extraction. The DenseNet was trained with CIFAR10 and CIFAR100 datasets. The result reported an accuracy of 83.98% for CIFAR10 and 51.19% for CIFAR100 dataset respectively, then finally [75] proposed a Multi-Scale DenseNets (MS-DenseNet) for aircraft detection from remote sensing images. In the study, a Feature Pyramid Network (FPN) was applied for the extraction of multi-scale feature detection with a focus on the features of small objects. The result of the MS-DenseNet reported a recall of 94%, an F1-score of 92.7%, a training time of 0.168s and a detection time of 0.094s. Overall these studies have investigated the performance of DenseNet for object detection problems, considering diverse datasets. From the review, it was observed while high success was recorded for metrics such as accuracy, recall, mAP, the delay during object detection [87,88] may affect its reliability when deployed as the computer vision model [76] for guidance assistance system for blind navigation. In addition, the maps despite their success need to be improved. This is because precision is a crucial factor that shows how reliable a system is, and with a poor precision result, the detection output of the model may be compromised and affect the accuracy of the audio feedback mechanism, thus leading to a poor decision by the user which might lead to object collision and accident. In summary, while DenseNet is successful for object detection in images, it may not be the best for real-time classification problems. The summary of the literature review on DenseNet is presented in Table 3;

Table 3. Non-exhaustive summary for DenseNets.

Authors	Primary objectives	Methods	Findings	Research gap
[85]	Object detection	F-RCNN based DenseNet and SSD based DenseNet	mAP of 5.63 with F-RCNN-based DenseNet and 3.86 with SSD-based DenseNet	Overall results can be improved
[86]	Object detection	SDD-based DenseNet	mAP of 0.9854	Delay speed
[73]	Object classification.	RPN-based DenseNet; PASCAL VOC and MS COCO datasets	PASCAL VOC reported 80.30% mAP and MS COCO dataset reported mAP of 55.0%,	Overall results can be improved
[74]	Object classification.	Deep Pyramidal Residual Networks (DPRN) and DenseNet; CIFAR10 and CIFAR100	Accuracy of 83.98% for CIFAR10 and 51.19% for CIFAR100	Results need improvement
[75]	Object detection	Multi-Scale DenseNets (MS-DenseNet	Recall of 94%, an F1-score of 92.7%, a training time of 0.168s and a detection time of 0.094s	Overall results can be improved

4. Relevant Literatures on object detection with ResNet

Today, ResNet has been applied in many areas for object detection and image classification. For instance, it was proposed for underwater object detection in [77], using a Multi-scale-ResNet (M-ResNet). The multi-scale segment [75] was applied to allow the detection of underwater objects with small sizes. The trained model reported a mAP of 96.5%. In [89], the Detection Transformer (DETR) algorithm was applied to improve ResNet for end-to-end object detection. The DETR was aimed at improving object feature detection to provide an effective image representation task. The tested model on diverse objects reported an average precision of 0.82 and a mean average recall of 0.63, which is good, but was improved in [90] using a hybrid approach that combined YOLOV and ResNet.

 doi:10.20944/preprints202402.0583.v1

The ResNet was utilized as the backbone of the YOLOV which is responsible for feature extraction to pool the diverse feature maps of the image and concentrate in the neck of the YOLOV for object detection. The results reported mAP of 95.1% and an F1-score of 98%, while the training speed of ResNet classification was considered in [91] using the Kaggle indoor scenes dataset and reported 142.2 minutes with an accuracy of 74%. From the review of ResNet for object detection applications, it was observed that while standalone ResNet recorded good performance for accuracy, mAP and recall, the combination of YOLOV, or multiscale feature detection mechanism has the potential to improve the success of real-time object detection applications. The summary of the literature review on ResNet for object detection is reported in Table 4.

Table 4. Non exhaustive summary of literature on ResNet.

Authors	Primary objectives	Methods	Findings	Research gap
[77]	detection of underwater objects with small sizes	Multi-scale-ResNet (M-ResNet).	mAP of 96.5%.	Cannot detect object of instance
[89]	End to end object detection	Detection Transformer (DETR) algorithm was applied to improve ResNet	average precision of 0.82 and mean average recall of 0.63,	Overall result need improvement
[90]	Object detection	YOLOV and ResNet	mAP of 95.1% and F1-score of 98%	Occlusion was not addressed
[91]	Indoor object detection	ResNet, Kaggle indoor scenes dataset	Accuracy of 74%.	Results need improvement

2.3. Identified Knowledge Gaps in the Reviewed Deep Learning Techniques

From the literature reviewed, it was observed that many studies have applied diverse deep learning techniques in addressing classification problems, among the approaches is Alex.Net which has been trained with several datasets and evaluated. The results despite the success revealed certain gaps such as occlusion in [78], the need for improved sensitivity in [79], and delay when applied for real-time classification in [70, 75 and 80]. In the case of Mobile-Net, which has also been tested on several datasets, overall results need improvement in [15,16], lack of instructiveness was identified in [68, 69 and 84] as the gap. DenseNet [85, 73, 74 and 75] leaves room for improvement in overall results reported as the major gap, while ResNet in [89,91] needs improved results. Lack of instructiveness was identified in [77] as the gap, while [89] suffers occlusion problems despite the success.

2.4. Bridging the Gaps with Real-Time Object Detection Models

While the reviewed literature discussed how the various deep learning techniques were applied in many cases for object detection [77,78,85,89], through image classification, gaps such as lack of instructiveness, delay, occlusion, need or improved overall performance, are some of the main gaps identified. To bridge this gap, one-stage object detection models are specially tailored for real-time classification problems. According to [92] YOLOV series has consistently gained momentum as a popular object detection model, with advantages such as high performance results, speed of real-time classification and foundation to address other identified gaps in the deep learning-based literature reviewed. YOLOV is an object detection model for real-time classification problems, according to [93]. In Figure 6 a basic architectural block of YOLOV-5 [95] which is a popular YOLOV series was presented. Figure 6 presents the architecture of the YOLOV 5, with three main sections which are the backbone, neck and output or the head. The backbone is the part responsible for the feature extraction process using Darknet [95] as the pre-trained model, Batch Normalization, and Leaky ReLU (CBL), Spatial Pyramid Pooling (SPP) and Cross Stage Partial (CSP) for the feature extraction of input images, then the concatenation (concat) layer, and CSP and CBL fusion the images for training and prediction in the output with three layers which represents the image classified, bounding box and confidence score.

 doi:10.20944/preprints202402.0583.v1

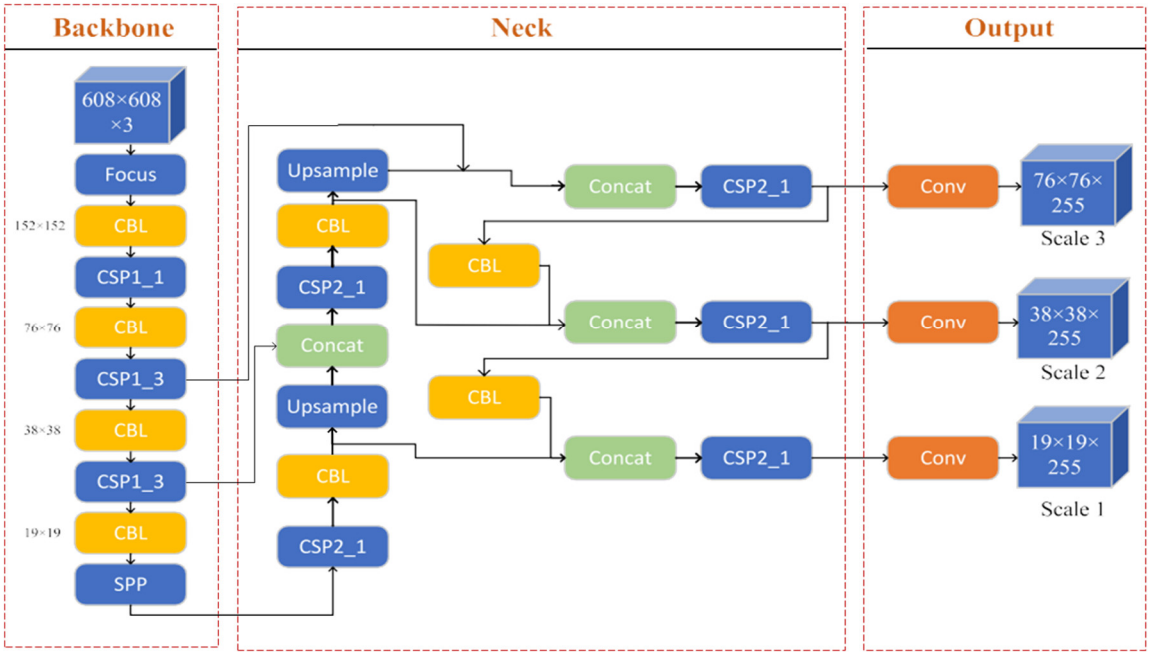

Figure 6. Architectural model of YOLO-5 [94].

2.5. Literature Review on the Application of YOLO for Real-Time Object Detection

Over the years, many studies have applied YOLOV for the classification of objects in real-time. For instance, in [95], Super-Resolution Reconstruction (SRR) was applied to optimize the performance of YOLOV-5. The SRR was focused on the image enhancement of small dense faces in real-time, while the YOLOV-5 solves the real-time classification problem. The wider face dataset used to train the model when tested reported a precision score of 88.2% and recorded a 2.9% improvement when compared to the standard YOLOV-5 algorithm. In another work [96] an improved performance (accuracy of 90.06%, mAP of 90.06%) was recorded when Repetitive Convolution (RepConv), Transformer Encoder (T-E), and Bil Bidirectional Feature Pyramid Network (BiFPN) modules were integrated into the architecture of YOLOV-5 and trained with UCAS-AOD and HRSC2016 datasets; while [97] applied Bidirectional Feature Pyramid Network (BiFPN) for multi-scale diverse feature map fusion in the neck of YOLOV-5.In addition, a Convolutional Block Attention Module (CBAM) was also applied, while a new non-maximum suppression technique using Distance Intersection Over Union (DIOU) was integrated into the model to address feature enhancement and bounding box overlaps. BDD100K dataset was used to train the model and a precision of 72.0%, recall of 42.1% and mAP of 49.4% were reported. Overall the literatures so far in this section are dominated by YOLOV-5 with various techniques employed to optimize the classification performance, from the study it was observed that [96] those who applied RepConv, T-E, and BiFPN modules to optimize YOLOV-5 recorded the best success rate for real-time object detection.

A voice assistance system for real-time object detection using YOLOV-4 was presented in [98]. The YOLO algorithm was trained with COCO dataset and deployed on Android application system. In addition, an algorithm was developed to compute objects of interest considering distance and then relay the output via sound to the user. The model mAP when tested using car and motorcycle as the objects of interest recorded 0.74 and 0.55 respectively. [99] Compared Fast Recurrent Convolutional Network (F-RCNN), YOLO-v4, YOLO-v4-hybrid, YOLO-v3, and SSD on COCO dataset considering frame captured per second (fps), speed and mAP. The results reported 110.56fps, 0.986mAP and 16.01ms or YOLOV-4 hybrid, as against the 60.2 fps yielded by YOLO-v4, 0.976 map, and 16.47ms detection speed; 61.3 fps by the YOLO-v3, 0.974 map, and 19.76ms detection speed; SSD's 5.8 fps, 0.883 map, and 178.6ms detection speed; F-RCNN's 3.7 fps, 0.925 map, and 275ms detection speed. [100] Applied YOLOV-3 on indoor object recognition problem and reported. In the model, CNN in the Yolov-3 was used to determine the bounding box class probability of objects. For the data acquisition process, OpenCV was applied and overall the classification accuracy was measured using

 doi:10.20944/preprints202402.0583.v1

accuracy and mAP which reported (99% and 100%) respectively, however, while this study [100] recorded a significant success, [101] argued that issues of over-fitting, varnishing gradient problem [16] due to small size of indoor dataset utilized may affect the model reliability. To address this challenge [101] applied DenseNet for feature connection and spatial separation convolution to replace the normal convolution. The aim is to apply these two strategies for parameter reduction and optimization of YOLOV-3. Comparative analysis with standard YOLOV-3 on ship image reported that the DenseNet-based YOLOV-3 recorded mAP of 0.96 as against 0.93 in YOLOV-3. In another study by [102], F-RCNN was applied to optimize the bounding box prediction in YOLOV-3 as a hybrid model. The model was trained with PASCAL VOC, VGG-M, and CaffeNet datasets and compared with the standard F-RCNN considering mAP. The results for each dataset with the hybrid YOLOV model reported (75.7, 64.2, and 62.2) respectively as against the (66.9, 59.2, and 57.1) obtained from the F-RCNN respectively, while in the same vein, [103] applied recall, and fps to measure compared the two models and recorded 0.451fps and 0.83mAP for YOLOV-3 as against 0.891, and 0.893 yielded by Faster RCNN. While these studies have all recorded significant contributions for real-time object detection problems, it was observed that YOLOV-3 when improved with F-RCNN [22,24] and DenseNet [16] when applied to improve YOLOV-3 has the potential to optimize the classification performance; however argued that while YOLOV-3 records high classification accuracy and mAP, that YOLOV-4 is better when compared with speed and mAP performance respectively. This suggests that the higher version of the YOLOV series supersedes the lowest version for real-time classification problems. The summary of the literature review on YOLOV for object detection is reported in Table 5.

Table 5. Summary of literature for object detection with YOLOV.

Author	Primary objective	Methods	Findings	Research gap
[95]	Real-time object detection	Super-Resolution Reconstruction (SRR) was applied to optimize the performance of YOLOV-5.	Precision score of 88.2%	Need to improve accuracy
[96]	Object detection while address overlap in bounding box	Repetitive Convolution (RepConv), Transformer Encoder (T-E), and Bil Bidirectional Feature Pyramid Network (BiFPN) modules were integrated in the architecture of YOLOV-5	Accuracy of 90.06%, mAP of 90.06%)	Need to improve accuracy
[97]	Object detection with YOLOV-5	Bidirectional Feature Pyramid Network (BiFPN) for multi-scale diverse feature map fussion and Convolutional Block Attention Module (CBAM) were integrated YOLOV-5	Precision of 72.0%, recall of 42.1% and mAP of 49.4%	Overall result despite the success need improvement
[98]	Interactive object detection system	YOLO algorithm was trained with COCO dataset	mAP of 0.74 and 0.55.	Result need improvement
[99]	Comparative deep learning application on object detection	Compared Fast Recurrent Convolutional Network (F-RCNN), YOLO-v4, YOLO-v4-hybrid, YOLO-v3, and SSD on COCO dataset considering frame captured per seconds (fps), speed and mAP.	Hybrid YOLOV-4 reported 110.56fps, 0.986mAP and 16.01ms as the best model	High delay and may not be reliable for real-time classification
[100]	Indoor object detection	Applied YOLOV-3 for indoor object detection	Accuracy and mAP, reported (99% and 100%)	Did not address issues of bounding box overlapping
[101]	Address over-fitting, parameter overload, problem with YOLOV	DenseNet was applied for feature connection and spatial separation convolution for parameter	DenseNet based YOLOV-3	Occlusion was not addressed

 doi:10.20944/preprints202402.0583.v1

		reduction and optimization of YOLOV-3.	recorded mAP of 0.96.	
[102,103]	Address bounding box overlapping problem in YOLOV	F-RCNN was applied to optimize the bounding box prediction in YOLOV-3	mAP reported 67.36% for the improved YOLOV; [66] reported, 0.45fps, 0.83 for YOLOV-3, as against 0.891, and 0.893 yielded by Faster RCNN.	Overall the results need to be improved

2.6. Current Challenges of YOLOV for Object Detection

From the literature reviewed, YOLOV has been applied by many researchers for real-time object detection and recorded good performance when considering parameters such as object detection speed, accuracy, precision and recall. However in certain applications such as surveillance, and visually impaired navigation, where a specific object of interest is required for detection in clustered scenes, YOLOV suffers occlusion problems [101], poor performance in changing environments, poor performance on object detection with poor lightening conditions [105].

2.7. Review of Techniques to Address Occlusion and Objects in a Changing Environment

ROI is a crucial component in various applications of real-time image classification systems such as wearable devices for the visually impaired [106], autonomous vehicles [107], surveillance systems [108,109], industrial automation [110], and human-computer interaction [111]. This is necessary to facilitate the primary goal of the computer vision task, by identifying and localizing a particular object, which informs speedy decision-making. While deep learning algorithms have dominated basic object detection models, and form the foundation for the OOI, other approaches have been proposed to make the classification model output more object-specific. For instance [30] applied Class Activation Mapping (CAM) and Adaptive Offloading Algorithm (AOA) for the extraction of edge-assisted lightweight region-of-interest for vehicle perception. The CAN was used for Region of Interest (ROI) mapping, while the AOA was applied to prompt inference through the adjustment of the down-sampling rate of the boxes in vehicle-to-edge communication. The benchmark YOLOV-5 model was also integrated with ResNet in the backbone to facilitate feature extraction, while the ROI and AOA search for the interested object. The results when compared to Youtube video generated from in-vehicle cameras, reported a 16% improvement against standard YOLOV-5 and a transmission demand reduction of 49%. While this study recorded a great improvement in ROI, [33] argued that issues of occlusion especially when there are multiple objects of the same kind remained a major challenge. To address this problem, YOLOV was used for the object tracking algorithm, then Kalman filter was applied to estimate the object state, then spatio-temporal feature information between the object and environment was used for the multiple object tracking [34]. The results when tested on a multi-object tracking dataset with a detection threshold set to 0.5, reported an accuracy of 74.5% and when the detection threshold was set to 0.2, the tracking accuracy was 73%. However, this method cannot be applied to all systems like visually impaired navigation, where only one object of interest detection is required. In another study [41] applied grid virtual division which operates based on pixel grayscale values for high-speed extraction of OOI in an optical camera communication system. The grid division strategy divides the received image into blocks and randomly samples several pixels in different blocks to locate the OOI characterized by the high grayscale values in the original image. The result when tested reported a transmission frequency of 5kHz. However, the study did not consider occlusion when there are similar multiple objects in the same scene. Object occlusion detection algorithm was proposed in [114] using camera information calibration based on the Gaussian mixture model [115], depth estimator using projective matrix in 3D space, and occlusion region detector using estimated object depth. The result when tested showcased the ability to detect

 doi:10.20944/preprints202402.0583.v1

occlusion by background, occlusion by another object and occlusion without depth estimation. However, if the object position is slanted, it may affect the accuracy of the object depth information. [116] applied Hard Example Mining (HAM) and Augmented Policy Optimization (APO) Approach. The APO applied nine augmented approaches [117], for the policy which focuses on policies such as CutOut, Contrast, MixUP, CutMix, blur, contrast, Hue, grayscale, and brightness [118], which has a positive influence on the learning data performance. The HAM extracted the hard positive data generated in the model training process using a false positive detection rate to detect the occluded objects. The result when tested demonstrated the model's ability to detect occlusion, and also mAP of 90.49% accuracy for the YOLOV model. In [119], an adaptive Spatio-temporal context (STC)-based algorithm for online tracking is proposed by combining the Context-Aware formulation (CAF), Kalman filter, and Adaptive Model Learning Rate (AMLR). The CAF was context context-aware filter for tracking the object, while the Kalman filter was applied for the object state estimation. The AMLR computes the mean of the image frames as the target motion of the image framed changes and is used to update the targeted model; however, despite the OOI success, the system is not interactive and therefore limits its application diversity. Overall, while these studies have made significant contributions to OOI, [114] is a more preferred approach due to its consideration of diverse occlusion problems and addresses its impact on OOI. Therefore, this algorithm will be applied as the OOI techniques to optimize the proposed YOLOV algorithm for real-time object detection which facilitates free navigation by impaired vision persons.

2.8. Open Research Gaps Not Addressed in Object of Interest Detection Literature Reviewed

From the literature reviewed which addressed issues of occlusion and detection of objects in a challenging environment like areas with low light, objects whose behavior changes with time, the application of techniques such as Spatio-temporal context (STC)-based algorithm, Context-Aware formulation (CAF), Kalman filter, and Adaptive Model Learning Rate (AMLR), Gaussian Mixture Model, Class Activation Mapping (CAM) and Adaptive Offloading Algorithm (AOA) have all made significant contributions when detecting an object of interest, however in certain application such as wearable devices for visually impaired navigation, vehicle collision detection system, surveillance system, where one particular object of interest is required at a time to inform other decisions, solution has not been obtained to address this problem. For instance in visually impaired navigation or surveillance systems, while occlusion detection systems are crafted for multiple objects, of specific objects of interest as identified in the literature reviewed, in cases where multiple objects of the same kind are in one scene, solution have not been presented to address such problem. In addition, multiple objects of the same type, in a constantly changing environment are another problem unsolved and applies to accident detection and control systems for instance. Therefore there is a need for an occlusion object of instance detection technique capable of solving these problems.

3. Recommended Occlusion Based Object of Instance Detection (OBOID) Technique

Object of instance detection problem is complex as it involves multiple problems such as object detection, tracking in the case of dynamic environment, classification, occlusion detection and instant object mapping. Object detection and tracking can be achieved using YOLOV-5 with metrics like non-maximum suppression and intersection over union (IOU) and identified from the literature review [97] to be very efficient for real-time object detection and tracking problems. The detection object then undergoes post-processing using occlusion detection approach inspired by [99] which identifies the camera information, then state estimation of the classified objects from the camera using Kalman filter, and then spatial coordinate the bounding box for the selected classified object of interest to determine the specific object of instance and classify. However, in a case where there are multiple objects of interest detected, the spatial information for the camera and then proximity sensor spatial information of the object is used to determine the instant object of interest. The assumptions for the modeling of the OBOID are as follows; the YOLOV-5 was assumed to be adopted from [97] and then trained for real-time object detection and tracking system using an object detection dataset. The

 doi:10.20944/preprints202402.0583.v1

camera state and bounding box information were assumed from [99], while the object distance was computed using information from the proximity sensor.

3.1. Modeling Assumptions

The assumptions to integrate the classified object of interest information to the proximity sensor use spatial and time information of sensors. The camera spatial information includes bounding box coordinates, contour, and centroid coordinates, while the spatial information from sensors is distance, distances, position and orientation. The matching information is distance threshold and similarity in the time stamp. The recommended algorithm for the system integration was presented as algorithm 1, while the system flow chart is presented in Figure 6;

3.2. Integration Algorithm (Algorithm 1)

1. Start
2. Load object of interest detection model
3. Apply camera calibration to convert pixel information to coordinate
4. Collect object distance from environment using ultrasonic sensor
5. Align coordinated of classified images with ultrasonic measurement using time stamp and spatial information
6. Synchronize the time measurements and spatial information
7. Apply kalman filtering to combine classification output with distance measurements
8. Apply matching rules using time and distance threshold settings
9. Identify as the instance object of interest
10. End

Figure 7 demonstrates the process flow chart of the recommended OBOID. First, the object is detected and classified with the YOLOV model, before post-processing using the occlusion detection model is applied to identify the object of interest. This is achieved using camera information, estimation depth of the objects and a bounding box coordinated to measure and identifies the object of interest. However in a case where multiple object of interest is time and spatial information of the object classified and then that of proximity sensor was utilized as in the integration algorithm to determine the instance object of interest.

 doi:10.20944/preprints202402.0583.v1

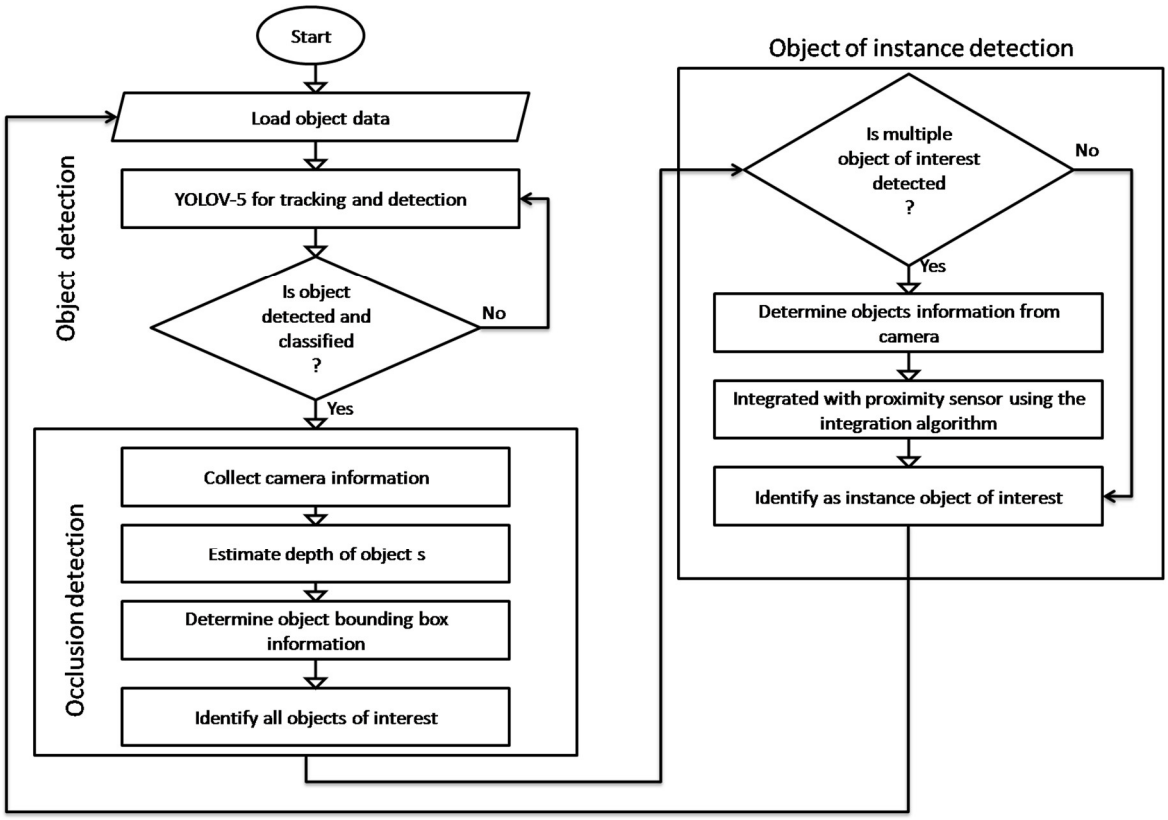

Figure 7. Recommended system flow chart for the OBOID.

3.3. Limitation of the OBOID

The choice of object detection functionality is dependent on applications; in this case, where only one object of instance is needed, the study is limited to applications such as impaired vision navigation guide systems, robotics vision, supervision and may not be necessary for a situation where the position of an object is not important.

4. Conclusion

This paper focused on literature applying deep learning techniques for image classification and object detection. Issues in the existing deep learning techniques when applied for object detection such as delay, poor accuracy, and issues of reliability were identified as gaps, and then one-stage object detection models such as YOLOV were investigated as proposed techniques to address the gaps. Issue of occlusion especially in objects habited within a dynamic environment, objects in poor lightening environment and objects of instance detection are some of the gaps identified with the current object detection model. A review of techniques to address occlusion and objects in changing environments was presented to identify solutions to these gaps, however, open gaps such as the object of instance detection model are not addressed. This research recommended an Occlusion Based Object of Instance Detection (OBOID) technique which used the spatial information from the camera and time to identify instant objects of interest in clustered scenes.

Reference

1. Alzahrani, N.; Al-Baity, H.H. Object Recognition System for the Visually Impaired: A Deep Learning Approach using Arabic Annotation. Electronics 2023, 12, 541. https://doi.org/10.3390/ electronics12030541
2. Chen, L.; Li, S.; Bai, Q.; Yang, J.; Jiang, S.; Miao, Y. Review of Image Classification Algorithms Based on Convolutional Neural Networks. *Remote Sens.* **2021**, *13*, 4712. https://doi.org/10.3390/rs13224712
3. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. arXiv 2013, arXiv:1311.2524

4. Toshev, A.; Szegedy, C. DeepPose: Human Pose Estimation via Deep Neural Networks. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 25 September 2014; pp. 1653–1660
5. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 2015, 39, 640–651.
6. Karpathy, A.; Toderici, G.; Shetty, S.; Leung, T.; Sukthankar, R.; Fei-Fei, L. Large-Scale Video Classification with Convolutional Neural Networks. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 25 September 2014; pp. 1725–1732.
7. Wang, N.; Yeung, D.Y. Learning a Deep Compact Image Representation for Visual Tracking. In Proceedings of the 26th International Conference on Neural Information Processing Systems—Volume 1, Lake Tahoe, NV, USA, 5–10 December 2013; NIPS'13. Curran Associates Inc.: Red Hook, NY, USA, 2013; pp. 809–817
8. Dong, C.; Loy, C.C.; He, K.; Tang, X. Learning a Deep Convolutional Network for Image Super-Resolution. In Computer Vision— ECCV 2014; Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T., Eds.; Springer International Publishing: Cham, Switzerland, 2014; pp. 184–199
9. Alagarsamy S., Rajkumar D., Syamala L., Niharika L., (2023), "An Real Time Object Detection Method for Visually Impaired Using Machine Learning," International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, pp. 1-6, doi: 10.1109/ICCCI56745.2023.10128388.
10. Bhattacharyya, S. A Brief Survey of Color Image Preprocessing and Segmentation Techniques. J. Pattern Recognit. Res. 2011, 1, 120–129.
11. Vega-Rodríguez, M.A. Review: Feature Extraction and Image Processing. Comput. J. 2004, 47, 271–272
12. . D, Z.; Liu, B.; Sun, C.; Wang, X. Learning the Classifier Combination for Image Classification. J. Comput. 2011, 6, 1756–1763
13. Xiaobing, H., Zhong, Y., Cao, L., and Zhang, L. (2017). Pre-Trained AlexNet Architecture with Pyramid Pooling and Supervision for High Spatial Resolution Remote Sensing Image Scene Classification. Remote Sens., 9(8), 848. https://doi.org/10.3390/rs9080848
14. Pan T., Huang H., Lee J., & Chen C., (2020) Multi-scale ResNet for real-time underwater object detection. Signal, Image and Video Processing https://doi.org/10.1007/s11760-020-01818-w
15. Wahab F, Ullah I, Shah A, Khan R., Choi A and Anwar M., (2022) Design and implementation of real-time object detection system based on single-shoot detector and OpenCV. Front. Psychol. 13:1039645. doi: 10.3389/fpsyg.2022.1039645
16. Cannata G. (2021) "Vanishing gradient problem in Deep neural networks; causes and possible solutions" source [http://www.towardsdatascinece.com/vanishing-gradient-in-deep-neural-network] accessed 1/16/2024
17. Wang, J.; Hu, X. Convolutional Neural Networks with Gated Recurrent Connections. IEEE Trans. Pattern Anal. Mach. Intell. **2022**, 44, 3421–3436.
18. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Computer Vision-ECCV 2016; Leibe, B., Matas, J., Sebe, N.,Welling, M., Eds.; Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
19. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Computer Vision-ECCV 2016; Leibe, B., Matas, J., Sebe, N.,Welling, M., Eds.; Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
20. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
21. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
22. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
23. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2980–2988.
24. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans. Pattern Anal. Mach. Intell. **2017**, 39, 1137–1149. Electronics **2023**, 12, 541 15 of 16
25. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. arXiv **2020** arXiv:2004.10934.
26. Sagana C. et al., (2021)"Object Recognition System for Visually Impaired People," 2021 IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER), Nitte, India, pp. 318-321, doi: 10.1109/DISCOVER52564.2021.9663608.

27. Z. Cai, Y. Ou, Y. Ling, J. Dong, J. Lu and H. Lee, "Feature Detection and Matching With Linear Adjustment and Adaptive Thresholding," in IEEE Access, vol. 8, pp. 189735-189746, 2020, doi: 10.1109/ACCESS.2020.3030183.
28. Reddy S., Khatravath P., Surineni N. and Mulinti R., "Object Detection and Action Recognition using Computer Vision," 2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS), Coimbatore, India, 2023, pp. 874-879, doi: 10.1109/ICSCSS57650.2023.10169620.
29. Au J., Reid D. and Bill A., "Challenges and Opportunities of Computer Vision Applications in Aircraft Landing Gear," 2022 IEEE Aerospace Conference (AERO), Big Sky, MT, USA, 2022, pp. 1-10, doi: 10.1109/AERO53065.2022.9843684.
30. Gao, X.; Wang, Z.; Wang, X.; Zhang, S.; Zhuang, S.; Wang, H. DetTrack: An Algorithm for Multiple Object Tracking by Improving Occlusion Object Detection. *Electronics* **2024**, *13*, 91. https://doi.org/10.3390/electronics13010091
31. Chen Y., Yang P., Zhang N., & Hou J., (2023) Edge-Assisted Lightweight Region-of-Interest Extraction and Transmission for Vehicle Perception. arXiv:2308.16417v1 [cs.MM] 31 Aug 2023
32. yu, S.-E.; Chung, K.-Y. Detection Model of Occluded Object Based on YOLO Using Hard-Example Mining and Augmentation Policy Optimization. *Appl. Sci.* **2021**, *11*, 7093. https://doi.org/10.3390/app11157093
33. Chen Y., Yang P., Zhang N., & Hou J., (2023) Edge-Assisted Lightweight Region-of-Interest Extraction and Transmission for Vehicle Perception. arXiv:2308.16417v1 [cs.MM] 31 Aug 2023
34. Gao, X.; Wang, Z.; Wang, X.; Zhang, S.; Zhuang, S.; Wang, H. DetTrack: An Algorithm for Multiple Object Tracking by Improving Occlusion Object Detection. *Electronics* **2024**, *13*, 91. https://doi.org/10.3390/electronics13010091
35. Hu X., Zhang P., Sun Y., Deng X., Yang Y.,&Chen L., (2022) High-Speed Extraction of Regions of Interest in Optical Camera Communication Enabled by Grid Virtual Division. Sensors 2022, 22, 8375. https://doi.org/10.3390/s22218375
36. Jung, J.; Yoon, I.; Paik, J. Object Occlusion Detection Using Automatic Camera Calibration for a Wide-Area Video Surveillance System. *Sensors* **2016**, *16*, 982. https://doi.org/10.3390/s16070982
37. Stauffer, C.; Grimson, W.E.L. Adaptive background mixture models for real–time tracking. In Proceedings of the 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Fort Collins, CO, USA, 23–25 June 1999; Volume 2.
38. Ryu, S.-E.; Chung, K.-Y. Detection Model of Occluded Object Based on YOLO Using Hard-Example Mining and Augmentation Policy Optimization. *Appl. Sci.* **2021**, *11*, 7093. https://doi.org/10.3390/app11157093
39. Hataya, R.; Zdenek, J.; Yoshizoe, K.; Nakayama, H. Meta Approach to Data Augmentation Optimization. *arXiv* **2020**, arXiv:2006.07965
40. Mehmood, K.; Jalil, A.; Ali, A.; Khan, B.; Murad, M.; Khan, W.U.; He, Y. Context-Aware and Occlusion Handling Mechanism for Online Visual Object Tracking. *Electronics* **2021**, *10*, 43. https://doi.org/10.3390/electronics10010043
41. Hu X., Zhang P., Sun Y., Deng X., Yang Y.,&Chen L., (2022) High-Speed Extraction of Regions of Interest in Optical Camera Communication Enabled by Grid Virtual Division. Sensors 2022, 22, 8375. https://doi.org/10.3390/s22218375
42. Rodney L., Dong Z., & Mubarak S., (2017) ClusterNet: Detecting Small Objects in Large Scenes by Exploiting Spatio-Temporal Information. arXiv:1704.02694v2 [cs.CV] 4 Dec 2017
43. Li W., Zhou J., Li X., Cao Y., & Jin G., (2023) Few-shot object detection on aerial imagery via deep metric learning and knowledge inheritance. International Journal of Applied Earth Observation and Geoinformation 122 (2023) 103397https://doi.org/10.1016/j.jag.2023.103397
44. Santra, S., Mukherjee, P., Sardar, P., Mandal, S., Deyasi, A. (2020). Object Detection in Clustered Scene Using Point Feature Matching for Non-repeating Texture Pattern. In: Basu, T., Goswami, S., Sanyal, N. (eds) Advances in Control, Signal Processing and Energy Systems. Lecture Notes in Electrical Engineering, vol 591. Springer, Singapore. https://doi.org/10.1007/978-981-32-9346-5_7
45. Patrick L. (2023)"Deep learning based object detection in clustered scene; https://towardsdatascience.com/deep-learning-based-object-detection-in-crowded-scenes-1c9fddbd7bc4 [accessed 1/15/2023)
46. Kang, K.; Li, H.; Yan, J.; Zeng, X.; Yang, B.; Xiao, T.; Zhang, C.; Wang, Z.; Wang, R.; Wang, X.; et al. T-cnn: Tubelets with convolutional neural networks for object detection from videos. IEEE Trans. Circuits Syst. Video Technol. **2017**, 28, 2896–2907.
47. Ahmed, M.; Hashmi, K.A.; Pagani, A.; Liwicki, M.; Stricker, D.; Afzal, M.Z. Survey and Performance Analysis of Deep Learning Based Object Detection in Challenging Environments. Sensors **2021**, 21, 5116. https://doi.org/10.3390/s21155116
48. Kumar, V.; Azamathulla, H.M.; Sharma, K.V.; Mehta, D.J.; Maharaj, K.T. The State of the Art in
Deep Learning Applications, Challenges, and Future Prospects: A Comprehensive Review of Flood
49. Forecasting and Management. Sustainability **2023**, 15, 10543. https://doi.org/10.3390/su151310543

50. Kinoshita Y., Kiya H. (2019)" Convolutional neural network considering local and global features for image enhancement"; IEEE 19; International conference on image processing, Taiwan, 2019; pp. 2110-2114.
51. P.E. Kekong, I.A. Ajah., U.C. Ebere (2019). Real Time Drowsy Driver Monitoring and Detection System Using Deep Learning Based Behavioural Approach. International Journal of Computer Sciences and Engineering 9 (1), 11-21
52. Eneh P.C, Ene I.I., Egoigwe V.S. Ebere U.C. (2019). Deep Artificial Neural Network Based Obstacle Detection and Avoidance for a Holonomic Mobile Robot. International Research Journal of Applied Sciences, Engineering and Technology Vol.5, No.1; ISSN (1573-1405); p –ISSN 0920-5691 Impact factor: 3.5
53. Palwankar T., & Kothari K., (2022) Real Time Object Detection using SSD and MobileNet. International Journal for Research in Applied Science & Engineering Technology (IJRASET) ISSN: 2321-9653; IC Value: 45.98; SJ
54. Maurício, J.; Domingues, I.; Bernardino, J. Comparing Vision Transformers and Convolutional Neural Networks for Image Classification: A Literature Review. *Appl. Sci.* **2023**, *13*, 5521. https://doi.org/10.3390/app13095521
55. Taye, M.M. Theoretical Understanding of Convolutional Neural Network: Concepts, Architectures, Applications, Future Directions. *Computation* **2023**, *11*, 52. https://doi.org/10.3390/computation11030052
56. Qayyum, W.; Ehtisham, R.; Bahrami, A.; Camp, C.; Mir, J.; Ahmad, A. Assessment of Convolutional Neural Network Pre-Trained Models for Detection and Orientation of Cracks. *Materials* **2023**, *16*, 826. https://doi.org/10.3390/ma16020826
57. Stančić, A.; Vyroubal, V.; Slijepčević, V. Classification Efficiency of Pre-Trained Deep CNN Models on Camera Trap Images. *J. Imaging* **2022**, *8*, 20. https://doi.org/10.3390/jimaging8020020
58. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
59. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. *arXiv* **2015**.
60. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252.
61. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
62. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
63. Chollet, F. Xception: Deep learning with depth wise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
64. Tan, M.; Le, Q. Efficient net: Rethinking model scaling for convolutional neural networks. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114.
65. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6848–6856.
66. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
67. Honmote H., Katta P., Gadekar S., & Kulkarni M., (2022) Real Time Object Detection and Recognition using MobileNet-SSD with OpenCV. International Journal of Engineering Research & Technology (IJERT) http://www.ijert.org ISSN: 2278-0181 IJERTV11IS010070 www.ijert.org Vol. 11 Issue 01, January-2022
68. MathuraBai B., Maddali V., Devineni C., Bhukya I., & Bandari S., (2022) Object Detection using SSD-MobileNet. International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 09 Issue: 06 | Jun 2022 www.irjet.net p-ISSN: 2395-0072
69. Younis A., Shixin L., Shelembi J., & Hai Z., (2020) Real-Time Object Detection Using Pre-Trained Deep Learning Models MobileNet- SSD. © 2020 Association for Computing Machinery. ACM ISBN 978-1-4503-7673-0/20/01…$15.00 https://doi.org/10.1145/3379247.3379264
70. Kayadibi I., Güraksın G., Ergün U., &Süzme N., (2022) An Eye State Recognition System Using Transfer Learning: AlexNet-Based Deep Convolutional Neural Network. International Journal of Computational Intelligence Systems (2022) 15:49 https://doi.org/10.1007/s44196-022-00108-2
71. Wang X., Liu J., Zhang S., Deng Q., Wang Z., Li Y., & Fan J., (2021) Detection of Oil Spill Using SAR Imagery Based on AlexNet Model. Hindawi Computational Intelligence and Neuroscience Volume 2021, Article ID 4812979, 14 pages https://doi.org/10.1155/2021/4812979

72. M. Wang, S. Zheng, X. Li and X. Qin, "A new image denoising method based on Gaussian filter," 2014 International Conference on Information Science, Electronics and Electrical Engineering, Sapporo, Japan, 2014, pp. 163-167, doi: 10.1109/InfoSEEE.2014.6948089.
73. Li J., Chen W., Sun Y., Li Y., & Peng Z., (2019) Object Detection Based on DenseNet and RPN. Proceedings of the 38th Chinese Control Conference July 27-30, 2019, Guangzhou, China
74. Yin L., Hong P., Zheng G., Chen H.,&Deng W., (2022) A Novel Image Recognition Method Based on DenseNet and DPRN. Appl. Sci. 2022, 12, 4232. https://doi.org/10.3390/app12094232
75. Wang Y., Li H., Jia P., Zhang G., Wang T., & Hao X., (2019) Multi-Scale DenseNets-Based Aircraft Detection from Remote Sensing Images. Sensors 2019, 19, 5270; doi:10.3390/s19235270www.mdpi.com/journal/sensors.
76. Xu, D.; Wu, Y. Improved YOLO-V3 with DenseNet for Multi-Scale Remote Sensing Target Detection. *Sensors* **2020**, *20*, 4276. https://doi.org/10.3390/s20154276
77. Hindarto D., (2023) Battle Models: Inception ResNet vs. Extreme Inception for Marine Fish Object Detection. Sinkron :Jurnal dan Penelitian Teknik Informatika Volume 8, Number 4, October 2023 DOI : https://doi.org/10.33395/sinkron.v8i4.13130
78. Omar F., Abdulrazzaq S., & Jasim M., (2020) Design and implementation of image-based object recognition. Periodicals of Engineering and Natural Sciences ISSN 2303-4521 Vol. 8, No. 1, February 2020, pp.79-88
79. Rani S., Ghai D., Kumar S., Kantipudi P., Alharbi A., & Ullah M., (2022) Efficient 3D AlexNet Architecture for Object Recognition Using Syntactic Patterns from Medical Images. Hindawi Computational Intelligence and Neuroscience Volume 2022, Article ID 7882924, 19 pages https://doi.org/10.1155/2022/7882924
80. Gilan A., Emad M., & Alizadeh B., (2019) FPGA-based Implementation of a Real-Time Object Recognition System using Convolutional Neural Network. DOI 10.1109/TCSII.2019.2922372, IEEE Transactions on Circuits and Systems II: Express Briefs
81. Hiddir S., Cetin T., Musa Y., (2022)" Detection of invisible cracks in ceramic materials using by pre-trained deep convolutional neural network" Neural Computing and Applications 34(2477) DOI: 10.1007/s00521-021-06652-w
82. Yifeng Z., Deyun C. (2022)"Expression Recognition Using Improved AlexNet Network in Robot Intelligent Interactive System" **Internet of Robotic Things-Enabled Edge Intelligence Cognition for Humanoid Robots** "Volume 2022 | Article ID 4969883 | https://doi.org/10.1155/2022/4969883
83. Nawfal J., and Mungur A. (2022)"Performance evaluation between tiny YOLOV-3 and MobileNet SSDv1 for object detection," IEEE, 4th Internaitonal conference on emerging trends in Electrical electronic and communication engineering, Mauritus, pp. 1-6
84. Rakkshab V., Priyansh S., Kevin P. (2021)"Comparison of yolov3., yolov5 and MobileNet SSD V2 for real time mask detection, IRJET, ISSN 2395-0056, pp. 1156-1160.
85. Chen B., Shen Y., & Sun K., (2020) Research on Object Detection Algorithm Based on Multilayer Information Fusion. Hindawi Mathematical Problems in Engineering Volume 2020, Article ID 9076857, 13 pages https://doi.org/10.1155/2020/9076857
86. Gangodkar D., & Vimal V., (2021) Video Object Detection Using Densenet-Ssd. Webology, Volume 18, Number 5, 2021 ISSN: 1735-188X DOI: 10.29121/WEB/V18I5/62
87. Jakubec, M.; Lieskovská, E.; Bučko, B.; Zábovská, K. Comparison of CNN-Based Models for Pothole Detection in Real-World Adverse Conditions: Overview and Evaluation. *Appl. Sci.* **2023**, *13*, 5810. https://doi.org/10.3390/app13095810
88. Sun, P.; Zhang, R.; Jiang, Y.; Kong, T.; Xu, C.; Zhan, W.; Tomizuka, M.; Li, L.; Yuan, Z.; Wang, C.; et al. Sparse R-CNN: End-to-End Object Detection With Learnable Proposals. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 14454–14463
89. Suherman E., Rahman B., Hindarto D., & Santoso H., (2023) Implementation of ResNet-50 on End-to-End Object Detection (DETR) on Objects. Sinkron :Jurnal dan Penelitian Teknik Informatika Volume 8, Number 2, April 2023 DOI : https://doi.org/10.33395/sinkron.v8i2.12378
90. Jung H., & Rhee J., (2022) Application of YOLO and ResNet in Heat Staking Process Inspection. Sustainability 2022, 14, 15892. https://doi.org/10.3390/su142315892
91. Ali H., Kabir S., & Ullah G., (2021) Indoor Scene Recognition using ResNet-18. International Journal of Research Publications (IJRP.ORG)IJRP 2021, 69(1), 148-154; doi:.10.47119/IJRP100691120211667
92. Oluwaseyi E., Martins E., Abraham E. (2023)"A comparative analysis of YOLOV-5 and YOLOV-7 object detection algorithms";Journal of Computing and Social informatics (Vol 2; No. 1; pp. 1-12.
93. Chourasia A., Bhojane R., Heda L. (2023)"Safety Helmet detection: A comparative analysis using YOLOV-4,5 and 7,"IEEE Xplore; International conference for advancement in Technology (ICONAT), Goa, India, pp. 1-8.
94. Liu, X., Li, G., Chen, W., Liu, B., Chen, M., Lu, S., 2022. Detection of dense citrus fruits by combining coordinated attention and cross-scale connection with weighted feature fusion. Appl. Sci. 12 (13), 6600.

http://dx.doi.org/10.3390/ app12136600, URL: https://www.mdpi.com/2076-3417/12/13/6600. Number: 13 Publisher: Multidisciplinary Digital Publishing Institute.

95. Xu Q., Zhu Z., Ge H., Zhang Z., & Zang X., (2021) Effective Face Detector Based on YOLOv5 and Superresolution Reconstruction. Hindawi Computational and Mathematical Methods in Medicine Volume 2021, Article ID 7748350, 9 pages https://doi.org/10.1155/2021/7748350
96. Cao F., Xing B., Luo J., Li D., Qian Y., Zhang C., Bai H., & Zhang H., (2023) An Efficient Object Detection Algorithm Based on Improved YOLOv5 for High-Spatial-Resolution Remote Sensing Images. Remote Sens. 2023, 15, 3755. https://doi.org/10.3390/rs15153755
97. Chen H., Chen Z., & Yu H., (2023) Enhanced YOLOv5: An Efficient Road Object Detection Method. Sensors 2023, 23, 8355. https://doi.org/10.3390/s23208355
98. Nazir, Z., Iqbal, W.M., Hamid, K., Muhammad, B. A.H., NAzir, A.M., Qurra-Tul-Ann, and Hussain, N. (2023). "VOICE ASSISTED REAL-TIME OBJECT DETECTION USING YOLO V4-TINY ALGORITHM FOR VISUAL CHALLENGED". Tianjin DaxueXuebao (Ziran KexueyuGongcheng Jishu Ban)/Journal of Tianjin University Science and Technology. ISSN (Online): 0493-2137. E-Publication: Online Open Access Vol:56 Issue:02:2023. DOI 10.17605/OSF.IO/APQYH
99. Rath, S., Priyadarshini, B.B.S., Patel, K.D., Patra, N., and Sahu, P.(2023). "A REAL-TIME HYBRID-YOLOV4 APPROACH FOR MULTICLASSIFICATIONAND DETECTION OF OBJECTS". Journal of Theoretical and Applied Information Technology15th June 2023. Vol.101. No 11. 2023 Little Lion Scientific. ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195.
100. Salam, H., Jaleel, H., and Hameedi, S. (2021). "You Only Look Once (YOLOv3): Object Detection and Recognition for Indoor Environment". Volume 7, Issue 6, 2021. DOI: 10.5281/zenodo.4906284
101. Zhelin L., Zhao L., Xu H., Pan M. (2020) "Lightweight ship detection method based on yolov3 and denseNet"Mathematical problems in Enineering; Hindawi, Vol. 20; ID 4813183;pp1-10.
102. Tirupataiah, U., Rao, N.K., Gokuruboyna, R.S., and Rao, S.S. (2019). "Real Time Object Detection in Images using YOLO". International Journal of Innovative Research in Science,Engineering and Technology (IJIRSET). (A High Impact Factor, Monthly, Peer Reviewed Journal). Visit: www.ijirset.com. Vol. 8, Issue 8, August 2019. ISSN(Online): 2319-8753. ISSN (Print): 2347-6710
103. Hossain, M., Rahman, A., and Ahmed, H. (2022). "Identifying Objects in Real-Time at the Lowest Framerate". International Journal of Research and Innovation in Applied Science (IJRIAS) |Volume VII, Issue VIII, August 2022|ISSN 2454-6194
104. Anand S., Singh V. (2012)"state o the art in visual object tracking" Informatica 31; publication at: https://www.researchgate.net/publication/269399115
105. Anand S., Singh V. (2011)" Robust object tracking under appearance change conditions based on Daubechies complex wavelet transform" Int. J. Multimedia Intelligence and Security, Vol. 2, Nos. at: https://www.researchgate.net/publication/264821985
106. Hussan, T.I.M., Saidulu, D., Anitha, P.T., Manikandan, A., and Naresh, P. (2022). "Object Detection and Recognition in Real Time Using Deep Learning for Visually Impaired People". International Journal of Electrical and Electronics Research (IJEER). Research Article | Volume 10, Issue 2 | Pages 80-86 | e-ISSN: 2347-470X
107. Flores-Calero, M.; Astudillo, C.A.; Guevara, D.; Maza, J.; Lita, B.S.; Defaz, B.; Ante, J.S.; Zabala-Blanco, D.; Armingol Moreno, J.M. Traffic Sign Detection and Recognition Using YOLO Object Detection Algorithm: A Systematic Review. *Mathematics* **2024**, *12*, 297. https://doi.org/10.3390/math12020297
108. Kumari, P., Mitra, S., Biswas, S., Roy, S., Chaudhuri, S.R., Ghosal, A., Dhar, P., and Majumder, A. (2021). "YOLO Algorithm Based Real-Time Object Detection". June 2021| IJIRT | Volume 8 Issue 1 | ISSN: 2349-6002.
109. Mohana and Aradhya, R.H.V. (2019). "Object Detection and Tracking using Deep Learning and Artificial Intelligence for Video Surveillance Applications". (IJACSA) International Journal of Advanced Computer Science and Applications,Vol. 10, No. 12, 2019 pp 517-530.
110. Chadalawada, K.S. (2020). "Real Time Object Detection and Recognition using deep learning methods". Master of Science in Computer Science.
111. Jiang, P., Ergu, D., Liu, F., Cai, Y., and Ma, B. (2022). "A Review of Yolo Algorithm Developments". ScienceDirect. Procedia Computer Science 199 (2022) 1066–1073. This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0)Peer-review under responsibility of the scientific committee of the The 8th International Conference on Information Technology andQuantitative Management (ITQM 2020 & 2021)10.1016/j.procs.2022.01.135
112. Zhang, Fan & Hu, Miao. (2018). Memristor-based Deep Convolution Neural Network: A Case Study. Available at: https://www.researchgate.net/publication/328091629_Memristor-based_Deep_Convolution_Neural_Network_A_Case_Study/citations
113. Hu X., Zhang P., Sun Y., Deng X., Yang Y.,&Chen L., (2022) High-Speed Extraction of Regions of Interest in Optical Camera Communication Enabled by Grid Virtual Division. Sensors 2022, 22, 8375. https://doi.org/10.3390/s22218375

114. Jung, J.; Yoon, I.; Paik, J. Object Occlusion Detection Using Automatic Camera Calibration for a Wide-Area Video Surveillance System. *Sensors* **2016**, *16*, 982. https://doi.org/10.3390/s16070982
115. Stauffer, C.; Grimson, W.E.L. Adaptive background mixture models for real–time tracking. In Proceedings of the 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Fort Collins, CO, USA, 23–25 June 1999; Volume 2.
116. Ryu, S.-E.; Chung, K.-Y. Detection Model of Occluded Object Based on YOLO Using Hard-Example Mining and Augmentation Policy Optimization. *Appl. Sci.* **2021**, *11*, 7093. https://doi.org/10.3390/app11157093
117. Xie, H.; Wu, Z. A Robust Fabric Defect Detection Method Based on Improved RefineDet. *Sensors* **2020**, *20*, 4260
118. Hataya, R.; Zdenek, J.; Yoshizoe, K.; Nakayama, H. Meta Approach to Data Augmentation Optimization. *arXiv* **2020**, arXiv:2006.07965
119. Mehmood, K.; Jalil, A.; Ali, A.; Khan, B.; Murad, M.; Khan, W.U.; He, Y. Context-Aware and Occlusion Handling Mechanism for Online Visual Object Tracking. *Electronics* **2021**, *10*, 43. https://doi.org/10.3390/electronics10010043