iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 Issue: IV Month of publication: April 2022

DOI: https://doi.org/10.22214/ijraset.2022.41438

Review on Different Methods for Real Time Object Detection for Visually Impaired

Melisa Andrea Soans[1], Zara Qureshi[2], Vikas Gupta[3], Jyotsna More[4]
[1, 2, 3, 4]*Department of Information Technology, Xavier Institute of Engineering Mumbai, Mahim Causeway, Mahim (West), Raheja Hospital Marg, Mumbai, Maharashtra 400016, India*

Abstract: Real-time object detection is the task of doing object detection in real-time with fast inference while main- taining a base level of accuracy. Real time object detection helps the visually impaired detect the objects around them. Object detection can be done using different models such as the yolov3 model and the ssd mobilenet model. This paper aims to review and analyze the implementation and performance of various methodologies for real time object detection which will help the visually impaired. Each technique has its advantages and limitations. This paper helps in the review of different methods and help in selecting the best method for object detection.

Keywords: Tensor flow lite, MS-coco, Raspberry pi, object detection, gtts,

I. INTRODUCTION

There are huge number of visually impaired people, may be partially or fully blind. According to the World Health Organization (WHO) [1], the number of people of all ages visually impaired is estimated to be 285 million, among which 39 million are blind, this is global statistics. These people suffer many problems during their day to day activities. Real time object detection helps in detecting objects This paper provides information on the study of different methods for object detection.

A. Real-Time Object Detection on Mobile Devices via Compression-Compilation Co-Design [4]

The fast development and wide utilization of object detection techniques have aroused attention on each the accuracy and speed of object detectors. However, the present progressive object detection works are either accuracy-oriented employing a giant model however resulting in high latency, or speed-oriented using a light-weight model but sacrificing accuracy. during this work, we have a tendency to propose the YOLObile framework, a period of time object detection on mobile devices via compression-compilation co-design. a unique block-punched pruning theme is planned for any kernel size

Framework Design

1) Block-Punched Pruning: To achieve the first objective in Section 3, we propose a novel pruning scheme–block- punched pruning, which preserves high accuracy while achieving high hardware parallelism. In addition to the 3×3 CONV layer, it can also be mapped to other types of DNN layers, such as 1×1 CONV layer and FC layer. Moreover, it is particularly suitable for high-efficient DNN inference on resource-limited mobile devices.
2) Reweighted Regularization Pruning Algorithm: Let Wi RM×N×Kh×Kw In the previous weight pruning algorithms, methods such as group Lasso regularization. However, it leads to either potential accuracy loss or requirement of manual compression rate tuning. Therefore, we adopt the reweighted group Lasso (Candes, Wakin, and Boyd 2008) method. The basic idea is to systematically and dynamically reweight the penalties. To be more specific, the reweighted method reduces the penalties on weights with larger magnitudes, which are likely to be more critical weights, and increases the penalties on weights with smaller magnitudes
3) Mobile Acceleration with a Mobile GPU-CPU Collaborative Scheme: To achieve the second objective in Section 3, we propose a GPU-CPU collaborative computation scheme to improve the computational efficiency of DNNs on mobile devices. It can be observed that the multi-branch architecture, are widely used in many state-of-the-art networks such as YOLOv4. Mobile devices has mobile GPU and mobile CPU, currently the DNN inference acceleration frameworks such as TFLite and MNN can only support.DNN inference to be executed on either the mobile GPU or the CPU sequentially, which leads to a potential waste of the computation resources. Based on the minimum of GPU-CPU parallel computing time Tpar and GPU-only computing time Tser, we can select the optimal executing device for branch 2. Note that the determination of execution devices for each branch structure in YOLOv4 is independent to other branch structures. Thus, the execution

devices for all branch structures in the network can be solved by greedy algorithm (Cormen et al. 2009). On the other hand, limited by the power and area, mobile GPUs usually have lower performance. For the less computational intensive operations, such as point-wise add operation and point-wise multiplication operation, mobile CPU performs similar or even faster speed compared with mobile GPU. Therefore, for the branch structures with nonCONV operations, either of CPU or GPU can be used for each branch depending on total computation time. Take the three final output YOLO head structures in YOLOv4 as an example, as shown in Fig 1. After transposing and reshaping the output from the last CONV layer in each branch, we still need several non-CONV operations to get the final output. We measure the total GPU and CPU execution times for the non-CONV operations in each branch and denote them as tg0, tg1, tg2 and tc0, tc1, tc2 respectively. The Ttotal denotes the total computing time for all three branches[4].

Note that the final output has to be moved to CPU sooner or later, so we do not count the data copying time into the total computation time. As a result, we select the combination that has the minimum total computation time as our desired computation scheme. Putting all together, our proposed GPU-CPU collaborative scheme can effectively increase the hardware utilization and improve the DNN inference speed.

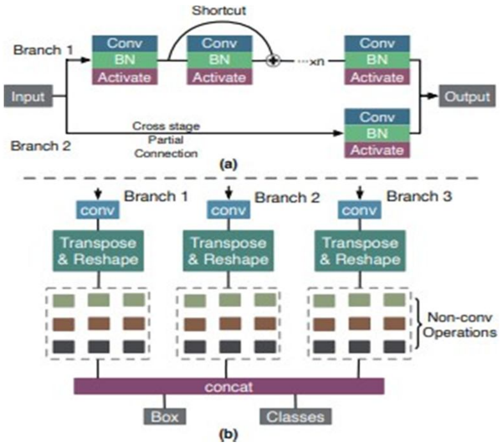

Fig. 1. An illustration of the (a) cross-stage partial block (b) non- convolutional operations in YOLO Head [4]

4) *Compiler-assisted Acceleration:* Inspired by PatDNN (Niu et al. 2020), YOLObile relies on several advanced compiler-assisted optimizations that are enabled by our newly designed block-punched pruning to further improve the inference performance. We summarize them here briefly due to the space constraints. First, YOLObile stores the model weights compactly by leveraging the pruning information (the block and punched pattern) that can further compress the index arrays comparing to the wellknown Compressed Sparse Row format. Second, YOLObile reorders blocks to improve memory and computation regularity, and to eliminate unnecessary memory access.
5) *Evaluation:* In this section we evaluate our proposed YOLObile framework on mobile devices in terms of accuracy and inference speed, compared with other state-of-the-art frameworks. Additionally, ablation study on different pruning schemes and configurations are provided.
6) *Evaluation of block-punched Pruning:* We first evaluate the accuracy and compression rate of our proposed block-punched pruning in YOLObile framework. As mentioned above, block size affects both accuracy and hardware acceleration performance. We adopt 8×4 as our block size, i.e. 4 consecutive channels of 8 consecutive filters.
7) *Evaluation of YOLObile Framework:* To validate the effectiveness of our framework, we compare our YOLObile with several representative works. To make fair comparisons, all the results (including the object detection approaches from the reference works) are evaluated under our compiler optimizations.
8) *Ablation Study:* Ablation study on pruning scheme. In this section, we conduct experiments on YOLOv4 under different pruning schemes

B. Real-time object detection and face recognition system to assist the visually impaired [1]

Accurate object detection from a real-time streaming video, relies heavily on machine learning. For a computer system to classify the objects present in a real-time video stream, it has to train with labelled data. The data consists of the labelled images for training, which requires us to have a large data set for training. Larger the data set, the greater is the accuracy of the trained model, hence we get better results during object detection.

Deep learning techniques making use of Convolutional Neural Networks (CNN) for object detection as shown in fig 2, are used extensively nowadays[1]. Neural networks consist of multi-layered architecture including an input layer, many hidden layers and an output layer. The input layer receives the extracted image of the object from the video stream. The hidden layers act as filters that receive an input from an upper layer, transform it with a specific pattern or feature and then send it to the next layer. The output layer classifies the image based on the input it receives from the hidden convolutional layers.

In Fast R-CNN shown, the entire complete image is processed to detect feature maps which are fed to the RoI layer to find region of interest in the given frame. The output given are probabilities of existence of the object in the frame which are then filtered based on some threshold value to give the final output. The training of all network layers is processed in a single-stage. It saves on storage space, and improves accuracy and efficiency with more appropriate training schemes.

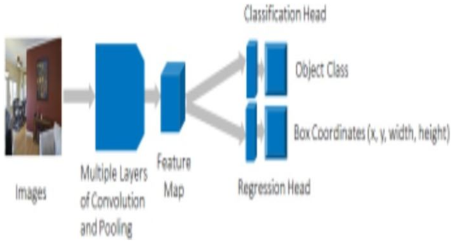
Classification Head
Object Class
Box Coordinates (x, y, width, height)
Images
Multiple Layers of Convolution and Pooling
Feature Map
Regression Head

Fig. 2. Convolutional Neural Network [1]

The time spent on every image is reduced to 10ms mak- ing this algorithm very fast. However, this approach involves a lot of computation making it infeasible for applications that need to run on CPUs with lesser computing power and limited computing resources such as embedded systems, smartphones, etc. Real time object detection using YOLO To overcome the drawbacks of the above algorithms, it is better to use a regression/classification based framework rather than the region proposed based frameworks mentioned above.

YOLO divides the input image into an S × S grid and each grid cell is responsible for predicting only one object. Each grid cell predicts a maximum of B bounding boxes and their corresponding confidence scores [1] as seen in fig 3.

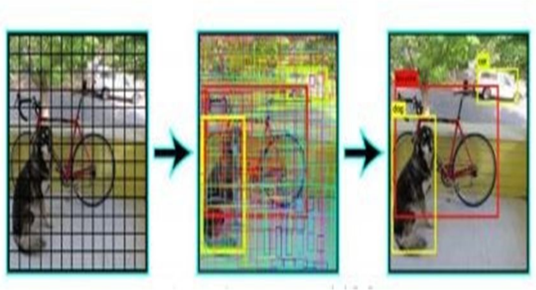

Fig. 3. YOLO Model [1]

The YOLO network can process images in real-time at 45 FPS and a simplified version Fast YOLO can reach 155 FPS with better results than other real-time detectors.

Advantages of YOLO

1) YOLO accesses the whole image in predicting boundaries while region proposal methods restrict the classifier to a specific region.

2) This method is fast and suitable for processing in real- time.
3) From a single network, object locations and classes are predicted. To enhance accuracy, end to end training is carried out.
4) YOLO is generalised and better compared to other methods when used in domains such as artwork.
5) YOLO demonstrates fewer false positives on background.
6) YOLO detects one object per grid cell while enforcing spatial diversity during prediction.

All of the improvements ensure that YOLOv2 is faster that YOLOv1 and has a higher degree of accuracy in object detection.YOLOv3 is another incremental improvement to the YOLO framework. On a Pascal Titan X, it processes images at 30 FPS and has a mAP of 57.9 percentage using the COCO dataset . Using this version of YOLO, real-time object detection for various applications can be accurately performed[1].

C. *Real Time Object Detection and Recognition for Blind People [3]*

Blind assistance is promoting a widely challenge in computer vision such as navigation and path finding. In this paper, two cameras placed on blind person’s glasses, GPS free service and ultra-sonic sensor are employed to provide. the necessary information about the surrounding environment. A dataset of objects gathered from daily scenes is created to apply the required recognition. Objects detection is used to find objects in the real world from an image of the world such as faces, bicycles, chairs, doors, or tables that are common in the scenes of a blind [3].

In this paper, two cameras placed on blind person’s glasses, GPS and ultra-sonic sensor are employed to provide the necessary information about the surrounding environ- ment. A dataset of objects gathered from daily scenes is created to apply the required recognition. The proposed method for the blind aims at expanding possibilities to people with vision loss to achieve their full potential [3].

The two cameras are necessary to generate the depth by creating the disparity map of the scene, GPS service is used to create groups of objects based on their locations, and the sensor is used to detect any obstacle at a medium to long distance. The descriptor of the Speeded-Up Robust Features method is optimized to perform the recognition. The proposed method for the blind aims at expanding possibilities to people with vision loss to achieve their full potential. The experimental results reveal the performance of the proposed work in about real time system.

The dataset contains photos of 91 objects that would be easily recognizable. Objects that are labeled using per- instance segmentation to aid in precise object localization. With a total of 2.5 million labeled instances in 328k images, the creation of the dataset drew upon extensive crowd worker involvement. Novel user interfaces for category detection, instance spotting and instance segmentation are also used [3].

As a dynamically typed language, Python is really flexible. This means there are no hard rules on how to build features, and you’ll have more flexibility solving problems using different methods (though the Python philosophy encourages using the obvious way to solve things). Furthermore, Python is also more forgiving of errors, so you’ll still be able to compile and run your program until you hit the problematic part.

The Raspberry Pi is a series of small single-board computers developed in the United Kingdom by the Raspberry Pi Foundation to promote the teaching of basic computer science in schools and in developing countries. The original model became far more popular than anticipated, selling outside of its target market for uses such as robotics. Peripherals (including keyboards, mice and cases) are not included with the Raspberry Pi. Some accessories however have been included in several official and unofficial bundles.

Fig. 4. Raspberry pi 3 model B [3]

The organization behind the Raspberry Pi now consists of two arms. Originally developed under the auspices of the Raspberry Pi Foundation, the success of the Pi Model B prompted the Foundation to set up Raspberry Pi Trading, with Dr Eben Upton as CEO, to develop the third model, the B+. Raspberry Pi Trading is responsible for developing the technology while the Foundation is an educational charity that exists to get that message out to schools. Raspberry Pi Trading reinvests about a third of its profit in RD, and the rest goes to the foundation.

The Microsoft Common Objects in Context (MS COCO) dataset contains 91 common object categories with 82 of them having more than 5,000 labeled instances, Figure (4.7) In total the dataset has 2,500,000 labeled instances in 328,000 images. In contrast to the popular ImageNet dataset [3], COCO has fewer categories but more instances per category. This can aid in learning detailed object models capable of precise 2D localization. The dataset is also significantly larger in number of instances per category than the PASCAL VOC and SUN datasets. Additionally, a critical distinction between our dataset and others is the number of labeled instances per image which may aid in learning contextual information.

The properties of the Microsoft Common Objects in Context (MS COCO) dataset in comparison to several other popular datasets. These include ImageNet , PASCAL VOC 2012 and SUN . Each of these datasets varies significantly in size, list of labeled categories and types of images. ImageNet was created to capture a large number of object categories, many of which are fine-grained. SUN focuses on labeling scene types and the objects that commonly occur in them. Finally, PASCAL VOC's primary application is object detection in natural images. MS COCO is designed for the detection and segmentation of objects occurring in their natural context.

Fig.5. Samples of images in the MS COCO dataset [3]

Detectors are convolutional filters, Each detector outputs a single value. discretizes the output space of bounding boxes into a set of default boxes over different aspect ratios and scales per feature map location. At prediction time, the network generates scores for the presence of each object category in each default box and produces adjustments to the box to better match the object shape. Additionally, the network combines predictions from multiple feature maps with different resolutions to naturally handle objects of various sizes[3]. Our SSD model is simple relative to meth- ods that require object proposals because it completely eliminates proposal generation and subsequent pixel or feature resampling stage and encapsulates all computation in a single network. This makes SSD easy to train and straightforward to integrate into systems that require a detection component.

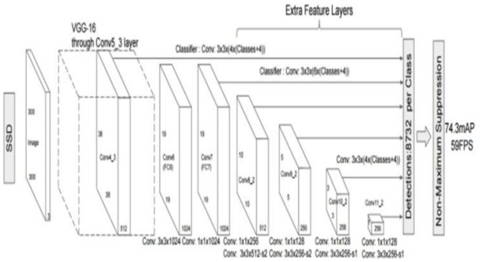
Extra Feature Layers
VGG-16
through Conv5_3 layer
Classifier : Conv: 3x3x(4x(Classes+4))
Classifier : Conv: 3x3x(6x(Classes+4))
SSD
300
Image
300
3
38
Conv4_3
38
512
19
Conv6
(FC6)
19
1024
19
Conv7
(FC7)
19
1024
10
Conv8_2
10
512
5
Conv9_2
5
256
3
Conv10_2
3
256
Conv11_2
256
Conv: 3x3x(4x(Classes+4))
Detections:8732 per Class
Non-Maximum Suppression
74.3mAP
59FPS
Conv: 3x3x1024 Conv: 1x1x1024 Conv: 1x1x256 Conv: 1x1x128 Conv: 1x1x128 Conv: 1x1x128
Conv: 3x3x512-s2 Conv: 3x3x256-s2 Conv: 3x3x256-s1 Conv: 3x3x256-s1

Fig. 6. Single Shot Detector SSD [3]

The view detection and recognition approaches have proven to work well in practice,after testing the project in- side the office in front of the blind person. It was detection and recognition correctly and in short time not exceeding two second.

D. Real Time Object Detection for Visually Impaired Person Using Tensor Flow Lite [5]

Real-Time Object Detection victimisation Tensor Flow fatless system has been developed to assist visually im- paired individuals with navigation and encompassing ob- jects detection. this method relies on raspberry pi, a single board cypher model, and also the Tensor Flow lite frame- work

Methodology

Workflow for implementation of item detected version is defined in Fig below three first raspberry pi is updated[5]. Next step is to put in dependencies for pi camera, then it's miles had to create surroundings. An surroundings is created so one can keep away from model complexities and to isolate bundle set up from the system. In that surroundings set up TensorFlow and open cv. The subse- quent step is to set TensorFlow version with dataset and last however critical step is take a look at results Image

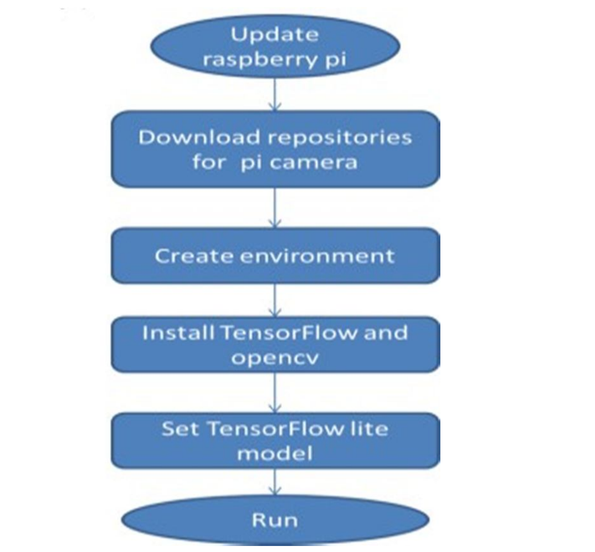
Update raspberry pi
Download repositories for pi camera
Create environment
Install TensorFlow and opencv
Set TensorFlow lite model
Run

Fig.7. Workflow [5]

Capturing is done with the help of Web camera or pi cam. The model takes input image. It is expected to have image with 300x300 pixels and there are 3 channels per pixel (Red, Green, Blue). and object detection model can identify which of a known set of objects might be present and provide information about their positions within the image. The detected object model has flattened buffer of 270,000 byte values (300x300x3) and model is quantised representing value between 0 to 255. A quantized neural network model is used which has an of 8-bit integer value, because it runs faster and speed up the actions. Frames per seconds (FPS) in Tensor Flow lite model is observed up to 4.4 which is comparatively more than TensorFlow which is 3. Output of model has 4 arrays (0,1,2,3), in which 0 represents location that is bounding box [top, left, bottom, right]. 1 represents classes it is as integer of 10 integers, indicating index of class label. 2 represents scores it is an array of 10 floating values between 0 and 1, as it is a probability to indicate class detection. 3 indicates number and detections it is an array of length 1 indicating total number of detection results. OpenCV supports the deep learning frameworks such as TensorFlow. OpenCV library is used for real time computer vision developed by Intel, includes statistical machine learning library which contains SVM(support vector machine), DNN(Deep neural network), K-NN, naïve neural network etc used for many real time applications like emotion recognition, face recognition, ob- ject detection, mobile robotics, motion tracking etc. refer section 1 for TensorFlow details. Google text to speech (gtts) is used here which uses (TTS) API. This library is used to read name of object detected. It can read unlimited number of letters and digits. Analysis

For testing the performance of developed model, it was tested under various scenarios. This includes • Variation in Light • Distance from camera • Background • Number of objects in frame There are 80 sample images in the data set, out of which few were tested, minimum 5183accurate results, this architecture can correctly label up to 5 objects in one capture. The conducted test showed that up to 10 to 12 feet, this is giving correct output for a number of objects in the range. Testing was carried out for various classes of same object for ex. Cell phone of 10 different companies were tested out of which 7 were detected accurately. Person was detected correctly till distance of 12 feet's.

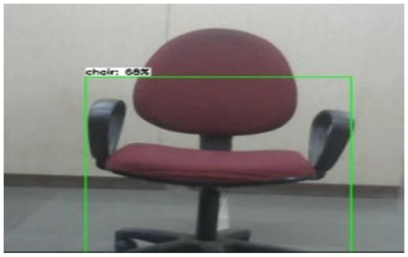

Fig. 8. Chair Detected at a distance of 68 percent [5]

E. Object Recognition for Visually Impaired using Machine Learning [2]

As a normal human being can use their eyes to look around them which helps them to detect and identify the object they are looking into. But it is a disadvantage for the visually impaired person as they cannot look around them to know what is going around and what is present in their surrounding environment. So, we have designed spectacles which will be a helping hand for the visually impaired person to know their environment much better like a normal human being can. We have used an algorithm known as You Only Look Once (YOLO). As the algorithm says, the camera which is present in the spectacle's observes the surrounding only once to identify the objects and the animals around[2].

Object Recognition plays a vital role in Artificial Intelli- gence. It is used in many of the upcoming technologies like the self-driving cars, video surveillance. The object recognition as some of the pre-built models in CNN, R- CNN. In object recognition the machine plays a vital role in finding the features of the objects under some of the prede- fined classes in the dataset. This classification is known as the object classification. Object Detection and classification is the two main building blocks of Artificial Intelligence. The Proposed system is to implement the technology of object recognition in the spectacles used by the visually impaired which will help them to detect the objects and the obstacles which is present in front of them. The system uses Raspberry Pi, Earphones, Miniature Camera, Wi-Fi Module and USB interface.

Here the camera is connected to the Raspberry Pi which detects the objects present in front of them. The camera captures the video.From the video the objects are detected and features of that object are considered and from the dataset class of the object is recognised based on those features and the object is detected. Now, after the recognition of the object is completed the name of the object is then converted to speech, so that it can be heard by the visually impaired person.

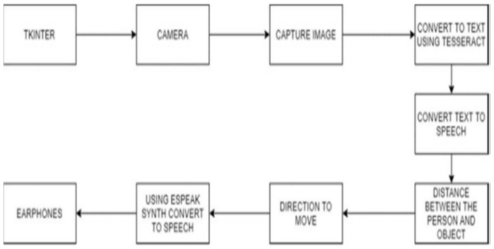
TKINTER
CAMERA
CAPTURE IMAGE
CONVERT TO TEXT USING TESSERACT
CONVERT TEXT TO SPEECH
DISTANCE BETWEEN THE PERSON AND OBJECT
DIRECTION TO MOVE
USING ESPEAK SYNTH CONVERT TO SPEECH
EARPHONES

Fig. 9. Flow-Diagram [2]

The above flow diagram displays the flow of data from the camera till the voice output is given to the visually impaired person.Here based on the recognized object the learning algorithm keeps on learning as soon as it recognises the objects and stores the recognised features and the x-axis, y-axis and the z-axis coordinates in the dataset so that if the same object is to be recognised in the future, then the same coordinates can be used to identify the object.

In YOLO once the image of the surrounding is captured, the objects in the surroundings are divided into regions, these regions have some predicting bounding boxes. The bounding boxes which are created have some particular weights that are weighed using the predicted probabilities.

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{obj} (x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2$$

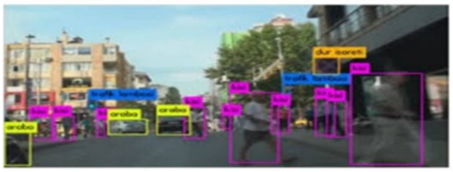

Fig. 10. Output of YOLO [2]

The implementation of the object recognition that would be helpful for the visually impaired person to guide them- selves and understand perfectly the object in front of them and the distance between the obstacle. Instead of the traditional methods that include walking canes and wheelchair which is difficult to use in public places. The proposed system provides much more accurate and more information than a conventional guiding system, as it can be used in different environments as it can be used in both indoors and outdoors[2]. By using this the visually impaired person can quickly become acquainted with their surroundings and can be prepared to react quickly in any circumstances which can occur at any time.

II. CONCLUSIONS

This paper mainly reviews the use of small deep neural network architectures for object detection such as Tiny YOLO to give good results and show that they can be used for real time object detection using mobile devices which can help the visually challenged.
It also reviews the proposed system which provides much more accurate and more information than a conventional guiding system, as it can be used in different environments as well as in both indoors and outdoors.

REFERENCES

[1] Anish Aralikatti*, Jayanth Appalla, Kushal S, Naveen G S, Lokesh S and Jayasri,"Real-time object detection and face recognition system to assist the visually impaired 1706 (2020)IEEE Conference on Computer Vision", The National Institute of Engineer-ing,Mysuru, India.

[2] Dr. Mamatha G1, Bharath Roshan B R2, Vasudha S R3,"Object Recognition for Visually Impaired using Machine Learning".International Journal for Research in Applied Science Engineering Technology (IJRASET) ISSN: 2321-9653.

[3] Redmon J, Divvala S, Girshick R and Farhadi A You Only Look Once: Unified, Real-Time Object Detection and Recognition for blind people 2016IEEE Conference on Computer Vision and Pattern Recognition (Las Vegas) pp 779-788

[4] Cai, Yuxuan Li, Hongjia Yuan, Geng Niu, Wei Li, Yanyu Tang, Xulong Ren, Bin Wang, Yetang. (2020). YOLObile: Real-Time Object Detection on Mobile Devices via Compression-Compilation Co-Design.

[5] Khandewale, Aditi, Vinaya V. Gohokar and Pooja Nawandar. "Real Time Object Detection for Visually Impaired Person Using Tensor Flow Lite." (2020).