Real-Time Object Detection with Live Camera

Deze Zhu, Wenlong Feng, Xiaoyi Lin
New York University, dz2372@nyu.edu, wf2142@nyu.edu, xl4708@nyu.edu

Abstract

This study presents an innovative application for real-time object detection and depth estimation using live camera feeds, particularly for autonomous vehicles. Utilizing Deep Neural Network (DNN) technology, specifically MobileNet SSD within the Caffe framework, the application achieves accurate identification and distance estimation of various objects. This enhances vehicle perception and navigation, contributing significantly to autonomous vehicle technology. The paper focuses on the development of object detection methods using MobileNet SSD, with a detailed exploration of its efficiency and accuracy in real-time scenarios. It discusses the relevance and advancement of DNNs in image recognition, highlighting the potential of combining MobileNet and SSD for practical applications in autonomous driving and traffic management. Our study also heightens the importance of knowing the distance information with the object detection, and how it could be feature combined with other fields and used in some real world scenario.

I. Introduction

In the realm of autonomous navigation and smart transportation, the integration of live-camera applications marks a significant stride. With the live camera, auto transportation can “see” the objects around it and use this information to deal with the situation to avoid any potential danger. Thanks to the fast development of deep learning technology, auto cars can “see more clearly” by driving more features from the camera including depth, the obstacle types or so on with a higher accuracy. As a result, in the rapidly evolving fields of computer vision and autonomous driving, the demand for real-time video-based object detection and recognition systems is intensifying.

Our project is set to pioneer an application that leverages such technology, enabling real-time object recognition and depth estimation through live camera feeds. This innovation is particularly envisioned for autonomous vehicles, enhancing their perceptual and navigational capabilities.

By employing Deep Neural Network (DNN) technology with the implementation of MobileNet SSD within the Caffe framework, our application aims to accurately identify various objects encountered by the vehicle and ascertain their distance from the camera. This dual capability not only enhances the vehicle's understanding of its immediate environment but also plays a critical role in ensuring safe and efficient navigation.

The potential applications of this technology are extensive, particularly in autonomous driving where the precise detection of objects and their respective depths is paramount for decision-making processes. Through this project, we aspire to contribute to the advancement of autonomous vehicle technologies, making them more reliable, responsive, and safe for real-world applications. The integration of this live-camera app into autonomous vehicles represents a significant step forward in the pursuit of fully autonomous and safe transportation solutions.

In our project, we mainly focus on the development of the object’s detection method using MobileNet SSD, and regard little on the smart cars’ computer decision parts on how to make right decisions.

II. literature survey

Live video object detection in autonomous vehicles is crucial for navigating complex road environments and identifying potential obstacles (Boukerche and Hou, 2021). This technology, crucial under varying environmental conditions, allows vehicles to detect and classify objects in real-time, providing essential information for safe navigation. By constantly analyzing live video feeds, autonomous vehicles can respond promptly to dynamic road scenarios, such as pedestrian movements, other vehicles, and unexpected obstacles. This capability significantly enhances the vehicle's decision-making process, ensuring safer driving experiences and reducing the risk of accidents. Moreover, it supports the development of more advanced and reliable autonomous driving systems, adapting to diverse environmental challenges and improving overall road safety.

In contemporary object detection research, there's a noticeable shift towards recognizing the significance of video data over static images. Zou et al.(2023) pointed out that this transition acknowledges the limitations of traditional image-based object detection methods, which often overlook the dynamics between video frames. Emphasizing the importance of spatial and temporal correlations in videos can

lead to more accurate and context-aware object detection systems. This approach is vital for applications where understanding the continuity and evolution of scenes is crucial, such as in autonomous vehicle navigation and surveillance systems.

In the field of autonomous vehicles, the importance of accurate distance measurement is paramount. Zaarane et al. (2020) address this by introducing a stereo vision-based system for measuring inter-vehicle distances. Their innovative approach combines vehicle detection with stereo matching, demonstrating high accuracy in distance calculation. This method, tested extensively, shows promise for real-time applications in autonomous driving and traffic management (Zaarane et al., 2020). This research is crucial for enhancing the safety and efficiency of autonomous vehicle navigation.

In the field of image recognition, Deep Neural Networks (DNNs) have revolutionized the way we understand and process visual information. And DNN has shown great performance on tasks like object detection according to Szegedy, Toshev, and Erhan(2013). Their ability to learn hierarchical representations from vast amounts of data has led to unprecedented advancements. DNNs can detect subtle patterns and features in images, making them ideal for tasks like facial recognition, object detection, and scene classification. They continuously improve with more data, leading to more accurate and nuanced image understanding. This technology has broad applications, from enhancing security systems to powering autonomous vehicles and aiding medical diagnostics. The continued development of DNNs in image recognition promises even more sophisticated and reliable visual processing capabilities in the future.

One of the classical object detection models is You Only Look Once (YOLO). The paper by Redmon et al. (2016) introduces YOLO, a novel framework for object detection that unifies the detection process in a single neural network. This approach significantly enhances real-time object detection speed while maintaining accuracy. YOLO's unique capability to look at the whole image during detection differentiates it from other region proposal-based methods, leading to faster and more efficient detection. Its impact on the field of computer vision, especially in applications requiring real-time processing, marks a significant advancement in object detection methodologies.

There are other methods for object detection. Doe's (2021) article provides an in-depth overview of MobileNet, a highly efficient convolutional neural network designed for mobile vision applications. The article delves into the structure of MobileNet, focusing on deep convolution (both deep and pointwise), and discusses MobileNet's architecture, parameters, and computational advantages of using width and resolution multipliers. The efficiency of MobileNet compared to other models amply emphasises its greater effectiveness with fewer parameters and operations. This makes MobileNet particularly suitable for applications with limited computational resources.

Liu et al. (2016) presented the SSD model, a method for object detection in images. This model is notable for its efficiency and accuracy, eliminating the need for bounding box proposals and pixel resampling. The SSD's architecture involves applying small convolutional filters to feature maps for prediction at multiple scales, thus handling objects of various sizes effectively. The paper details the model's structure, training methods, and performance evaluations, demonstrating its superiority in speed and accuracy compared to other methods like Faster R-CNN and YOLO, especially in real-time applications.

In the real world, researchers always use MobileNet and SSD together for object detection problems. The study by Younis et al. (2020) investigates the efficacy of the MobileNet-SSD model for real-time object detection, emphasizing its performance in surveillance scenarios. They highlight how this model combines the efficiency of MobileNet with the accuracy of SSD, making it well-suited for both indoor and outdoor environments where rapid and reliable object detection is crucial. The study's findings indicate impressive precision in object detection, with the algorithm achieving an average precision (AP) of 99.76% for cars, 97.76% for persons, and 71.07% for chairs. This level of accuracy notably enhances behavior detection capabilities, meeting the demands for real-time detection essential in both indoor and outdoor daily monitoring scenarios.

Choudhari et al. (2021) conduct a thorough comparison of the YOLO and SSD MobileNet algorithms for object detection in their study, specifically in the context of surveillance drones. They meticulously evaluate both models, considering crucial factors like detection accuracy, processing speed, and the capability for real-time implementation. This comparative analysis is particularly valuable for practitioners and researchers in the field of aerial surveillance, as it provides essential guidance on choosing the most effective algorithm based on specific operational requirements and constraints.

After research, a DNN model using mobileNet-SSD is chosen for the experiment, and beyond the original object recognition which will tell what the object is, the depth information will also be a key feature to be derived.

III. Technical details

The implementation of the project's core model, which is MobileNet-SSD includes to part, MobileNet and SSD.

SSD. The Single Shot Detector (SSD) is a NN architecture designed for detection purposes - which means localization (bounding boxes) and classification at once.
SSD was first introduced with a VGG-16 net as its base network, which provided high-level features for classification

or detection. The figure 1 shows the original design of SSD with a VGG-16 net before it. Our project uses a MobileNet instead which is similar toVGG-16 net as a base network.

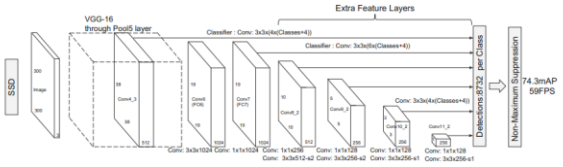

Figure1. Original SSD with VGG-16

SSD has some key features:

• **Fixed-Size Bounding Boxes:** It produces a set of fixed-size bounding boxes and scores for object class instances in those boxes, followed by non-maximum suppression for the final detections.

• **Default Boxes and Aspect Ratios:** SSD associates a set of default bounding boxes with each feature map cell. These boxes tile the feature map in a convolutional manner, and the network predicts offsets relative to these default box shapes, as well as class scores. Take Figure 2 as an example, the blue rectangles in the right photo shows how SSD use its default boxes and aspect ratios to extract location information and decided if there is an object on that location, and then it will tell what the object is with the pre-trained base net, and it will pick the one with most possibility as the final result showing in the camera view.

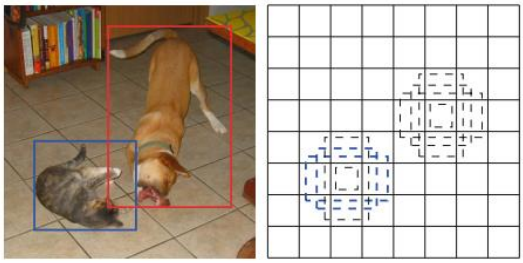

Figure 2. SSD boxes and ratios

• **Base Network Architecture & Additional Network Structure**: The early layers of the network are based on a standard architecture used for high-quality image classification, referred to as the base network. VGG-16 is used in this case, but the model is adaptable to other networks. To facilitate detection, the model adds auxiliary structure to the network. This includes layers that allow for predictions at multiple scales and convolutional predictors for each feature layer.

• **Efficiency and Accuracy:** The SSD model is designed to be efficient and accurate, eliminating the need for proposal generation and resampling stages, simplifying training, and allowing for integration into systems needing a detection component. As is inferred from its name, with a single input of image, it can give a multiboxes result containing all the objects it finds in the image within a very short of time. Making it good for real-time object detection.

The loss function of the SSD model is a weighted sum of two components: localization loss and confidence loss. localization loss measures how far the predicted bounding boxes are from the actual object locations. It typically uses a smooth L1 loss, which is a less sensitive version of L1 loss to outliers. And confidence loss assesses how well the model classifies objects within the bounding boxes. It usually involves a softmax loss over multiple classes, including a "no object" class. The total loss function is as below:

$$L(x,c,l,g) = (1/N) * \left(L_{conf}(x,c)\right) + \alpha L_{loc}(x,l,g) \quad (1)$$

Where x is Indicator for matching the default box to the ground truth box, c is Confidence loss, l is Predicted box, g is Ground truth box, N is Number of matched default boxes, α is weighting parameter balancing the localization and confidence losses.

MobileNet. Mobilenet net is a lightweight deep neural network proposed by Google for mobile phones and embedded scenarios. The main feature of Mobilenet network is that depthwise separable convolution is used to replace common convolution, thereby reducing the computing load. Improve computing efficiency of the network. The classification accuracy of the network on the ImageNet dataset reaches 70.8%, which greatly reduces the computation amount without losing much accuracy and makes it possible for the neural network model to run smoothly on the ordinary single chip computer.

MobileNet_SSD. The final work combines the MobileNet and SSD together for the object detection problem. The process includes training and testing. Also, a pre-trained will be used for the object detection project, after some necessary modification with the training net. The detailed information of the training testing and deploying are stored in the prototxt files, which is was invented by the developers of the Caffe deep learning framework. This format allows researchers and developers to rapidly experiment and tweak network designs without the need to write complex code.

the hyperparameters chosen for training and testing is in the solver file, and some important ones are introduced here: the train and test net specific the net configuration, base_lr, which is the starting learning rate is set to be 0.0005, and the policy for adjusting the learning rate is multistep in this model, also the step value is set to be '20000',40000'. These parameters are adjusted so the model could better deal with complex datasets.

After training and testing, a pre-trained mobileNet could be derived and by combining it with the proper SSD layers, the final mobileNet-SSD model can be built. Some of the crucial implementations are introduced here: the input configuration limits the input shape to be "1*3*300*300", one single image with 3 RGB colors and 300*300 resolution. Then some convolution layers (including Depthwise Separable Convolution such as conv1/dw and Standard Convolution with specific parameters like conv0) and activation layers (using ReLU like conv1/relu), some detailed setting including 'kernel_size' and 'stride' are set together with the

chosen layers. One thing to notice is that the Depthwise Separable Convolution is a special feature of MobileNet, where convolutions are split into depthwise and pointwise convolutions, reducing computational cost and model size. After that, some SSD Specific Layers like Mbox layers and PriorBox layers are added in (E.G. conv11_mbox_loc, conv11_mbox_conf, conv11_mbox_priorbox). Finally, there are Concatenation Layers, Output Reshaping and Softmax, and Detection Output Layer gathering the location predictions, confidence scores, and prior box information to determine the final detected objects.

Overall, this configuration combines the lightweight, efficient nature of MobileNet with the effective detection capabilities of SSD, making it suitable for real-time object detection tasks with limited computational resources.

The distance information has not been added to the application till now, so the final step is to add on the depth information of the object. This is realized by taking advantage of the Intel RealSense cameras and the actual depth was calculated with the helper function in the figure below which generates a matrix of floating-point values (in meters).

```
auto depth_mat = depth_frame_to_meters(pipe, depth_frame);
```

Figure 3. depth info helper function

IV. Experiment Result

There are two parts of the experiment, the first part is the model training and testing part in which we got the pretrained mobileNet-SSD model for future implementation.

The MobileNet-SSD model was first trained with the VOC0712 dataset and the mAP got was 0.68. However, if the model is fine-tuned with VOC0712 dataset after pretraining with MS-COCO dataset, the mAP can finally reach 72.7.

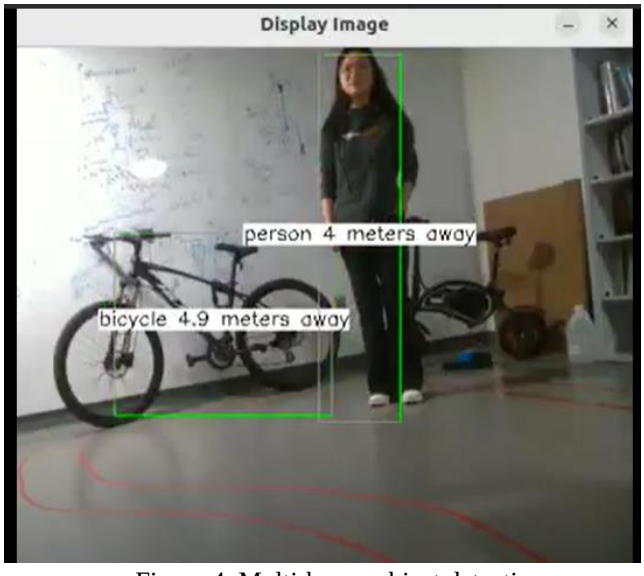

Figure 4. Multi-boxes object detection

After properly running the program with all environments installed, the program should run correctly, and the real-time video should be shown on the computer with boxes around detected objects. Figure 4 shows one frame of the real-time video where there are 2 main objects detected by the MobileNet-SSD with calculated distances. It is obvious that MobileNet-SSD successfully recognized these two objects and put boxes around them. Also, the depth information is also included on the screen. By telling with human insight, the person can be about 1 meter in front of the bicycle, it means the program has given a relatively correct result.

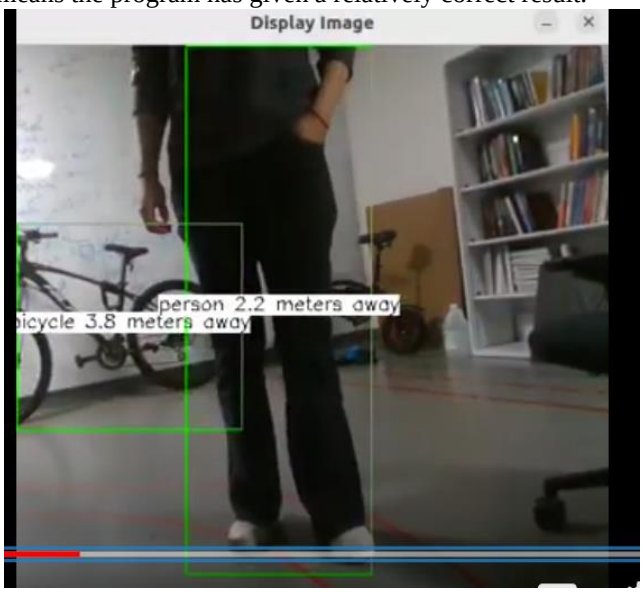

Figure 5. two objects with different positions

By looking at figure 5, when the person stepped a little bit forward and only part of the human was caught by the camera, the MobileNet-SSD can still tell there is a person there, maybe because the key feature of a human being can still be derived from this frame, and the estimated distance of the human is shorter than figure 4, which match the actual scene.

However, the application also has some problems, especially when two objects are overlapping each other. In figure 4, when the person and the bicycle are overlapping with each other, though MobileNet-SSD gave a correct recognition with proper round boxes, the estimated depth of the bicycle is different from that in figure 4. There are two reasons that are causing this problem. The first problem is that the camera has moved a bit with an angle, the objects near the side seemed to be closer to the camera than when it is in the center in a camera view. And the second problem is that the estimated distance was calculated using a mean distance of all points in the boxes, while the two objects are overlapping with each other, the distance from the human maybe treat as distance from the bicycle in some parts of the bicycle, and thus make the depth of bicycle smaller than the actual scene.

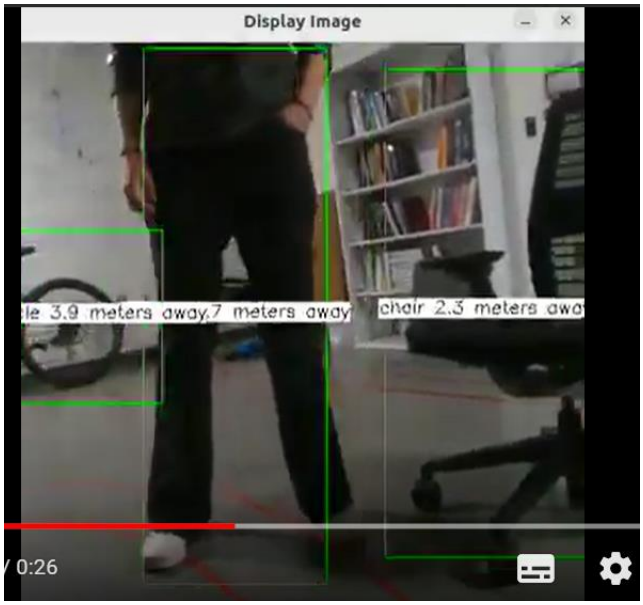

Figure 6. More objects

Furthermore, figure 6 demonstrated that the application still looks good with more objects and these objects are taking up most of the screen.

During the experiment, the video showing on the screen with boxes changes smoothly with little lagging. This is a very important key feature for the application because we are using a real-time camera and the in application could be future used with an auto-driven vehicle, the detection time needs to be short enough to save time for the vehicles to make follow decisions.

V. Conclusion and future work

To sum up, we were planning to build a real-time object detection system that can do recognize things quickly and correctly. In the initial plan, we hope the application can be future used with auto-driven cars to help them avoid obstacles. In the final work, our program can do objects detections with labels in the pretrained model, including most of the common things in the real world. The detection time is good enough with smooth screen video, and there seems to be no wrong detections. Also, the estimated distance information of the objects is calculated and tagged with the recognition round boxes.

Our code uses C++ with the necessary library installed to make good use of the camera and uses a MobileNet-SSD which combines the benefits of both MobileNet and SSD to perform a good final result.

It is clear that there are still some parts of the application that need to be improved, like how to calculate a more realistic depth information with overlapping things. Also, some necessary modifications are needed if we want to use it in any specific areas like auto-driven cars or medical area.

Reference

Boukerche, A. and Hou, Z., 2021. Object detection using deep learning methods in traffic scenarios. ACM Computing Surveys (CSUR), 54(2), pp.1-35.

Doe, J. (2021). An Overview on MobileNet: An Efficient Mobile Vision CNN. Available at: https://medium.com/@godeep48/an-overview-on-mobilenet-an-efficient-mobile-vision-cnn-f301141db94d [Accessed 10 October 2023].

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C., and Berg, A.C., 2016. SSD: Single Shot MultiBox Detector. In: ECCV 2016 Part I, LNCS 9905, pp. 21–37. Springer International Publishing. DOI: 10.1007/978-3-319-46448-0_2.

Phadtare, M., Choudhari, V., Pedram, R. and Vartak, S., 2021. Comparison between YOLO and SSD mobile net for object detection in a surveillance drone. International Journal of Science, Research and Engineering Management, 5, pp.1-5.

Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.

Szegedy, C., Toshev, A. and Erhan, D., 2013. Deep neural networks for object detection. Advances in Neural Information Processing Systems, 26.

Younis, A., Shixin, L., Jn, S., & Hai, Z. 2020, January. Real-time object detection using pre-trained deep learning models MobileNet-SSD. In Proceedings of 2020 the 6th international conference on computing and data engineering (pp. 44-48).

Zaarane, A., Slimani, I., Al Okaishi, W., Atouf, I. and Hamdoun, A., 2020. Distance measurement system for autonomous vehicles using stereo camera. Array, 5, p.100016.

Zou, Z., Chen, K., Shi, Z., et al., 2023. Object detection in 20 years: A survey. Proceedings of the IEEE.

Link To Our code: https://github.com/oliver1112/Real-Time-Object-Detection-with-Live-Camera