SMART WARNING SYSTEM USING EDGE COMPUTING

Devi. R, Information Technology, Government college of Technology, Anna University
India
r.devi@gct.ac.in

Abstract

In recent years, the occurrence of road accidents increases drastically. There are many factors influencing the road accidents. Nowadays usage of mobile phones while driving is common but it leads to the distraction in driving and subsequently causes accidents. Though jurisdictions are made that usage of electronic devices such as mobile phones, tablets, laptops while driving are illegal, no one is aware of that. The drivers who use mobile gadgets while driving not only risk their lives but also the people around them. Most of the road users are well aware of the general rules and safety measures. However due to human errors, accidents occur. To overcome from the human errors this paper focuses on developing a security alert system for drivers using edge computing paradigm.

Keywords: Avoid road accidents, edge computing, tensorflow image capturing

1 Introduction

The advancement in technology leads to the development of Internet of Things [1] (IoT) in every aspect of life. Large number of devices are interconnected with each other and communicate within themselves through network of distributed nodes. In this case, increasing number of IoT devices creates large amount of data. In Cloud computing, the data is processed only at the cloud data centers. For an IoT enabled system, the data fetched by the sensors and smart gadgets should be sent to the cloud server and computing will be done there and it should sent results again to the smart gadgets. However it leads to the high cost for data transmission and produces load in the network. CCSA defines IoT as A network which gathers information from the physical world objects such as sensors and gadgets by diverse deployed devices with the ability of perception, computation, execution and communication, as well as support communications among human and things or among things by transmitting, classifying, categorizing and processing information [2].

An IoT enabled security alert system for vehicle drivers must process the data quickly since it is a time sensitive situation even a microsecond delay in alert will lead to the accidents. The machine learning , a subset of Artificial intelligence can be used in IoT applications in order to make the application more effective and learn itself [3]. So the traditional cloud computing technique won’t be sufficient. Edge computing will be an effective solution. The data will be processed at the creation place and the burden of processing at the cloud centers will be greatly reduced. The computing power will be less compared to the cloud servers. However it will be sufficient for security alert system.

2 Existing systems for Collision detection

2.1 Forward Collision Warning (FCW)

The Forward Collision Warning (FCW) system uses radar to detect vehicles or obstacles in front of the car [4]. The system calculates the distance to the object in front and if the car gets close enough, it produces an alarm and displays a visual alert, prompting the driver to apply the brakes. Forward Collision Warning alerts if an object in the path suddenly stopped or slow down. FCW will significantly reduce the chance of a crash or a fatal accident as shown in figure 1. Drawbacks ----

![Figure 1: Forward Collision Warning System](image)

2.2 Automatic emergency braking (AEB)

AEB is the newer feature that alerts the driver to an imminent crash and assist the driver in using the maximum breaking capacity of the car. Three categories of AEB are Low speed system – this system is designed for use on city streets to detect other cars in front and to prevent crashes and minor injuries such as whiplash. Higher speed system – scans up to 200 meters in front of the car and uses a long range radar for faster speeds [5]. Pedestrian system – picks up pedestrian movement in relation to the direction your vehicle is traveling and determines risk of collision. Drawbacks of AEB are the system may not work when driving on a hill. It might not spot vehicles with high ground clearance or those with low rear ends. It may not
Object detection is the process of identifying the instances of a real world objects such as humans, cars, bikes, animals, gadgets and so on. It helps to capture multiple instances of an object within an image thereby getting additional information about the image. It is commonly used in applications such as image retrieval, security, surveillance, and Advanced Driver Assistance Systems (ADAS) [6]. The object detection can be done in two ways, Object detection on image and Object detection on real time. Some of the object detection techniques are, Feature based object detection, Viola-Jones object detection , SVM classifications with HOG features, Machine learning based object detection

3 Need of Object Detection

Object detection is the process of identifying the instances of a real world objects such as humans, cars, bikes, animals, gadgets and so on. It helps to capture multiple instances of an object within an image thereby getting additional information about the image. It is commonly used in applications such as image retrieval, security, surveillance, and Advanced Driver Assistance Systems (ADAS) [6]. The object detection can be done in two ways, Object detection on image and Object detection on real time. Some of the object detection techniques are, Feature based object detection, Viola-Jones object detection , SVM classifications with HOG features, Machine learning based object detection

3.1 Feature Based Object Detection

This method is based on the extraction of features from the contours of the images [7]. The overlapping contours and large contours are removed so that appropriate contour which can used in extraction .The features are extracted from those selected contours. Some of the features used are lines, ellipse, lobe and parallel lines. The extracted features are searched with the database which contains the features of the trained images. The feature with maximum probability are can be used to identify the objects within the captured images.

3.2 Viola-Jones Object Detection

Viola-jones algorithm is the Real-time face detection mechanism developed by Paul Viola and Michael Jones. It can be used to variety of objects but it is used primarily for face detection [8]. This algorithm uses Haar basis feature filters, so multiplication is not possible which leads to slow training of images .But this technique detects objects quickly. The objects should be shown straight without tilt otherwise the algorithm will not work properly.

3.3 SVM Classifications with Hog Features

In this method, Support Vector Machines (SVM) are used to train a model to classify if an image contains a particular object such as car or not [9]. The feature representation method Histogram of Oriented Gradients (HOG) are used as the feature representation. The small size of datasets is enough to predict the objects. In this method both the positive as well negative samples. A sliding window is used to detect objects in this method.

3.4 Machine Learning Based Object Detection

Machine learning based object detection not only detects the objects in the picture but also the localization of objects within the picture [10]. Localization refers to the place where object is detected within the picture. These machine learning based algorithms used in object detection can able to detect multiple objects simultaneously which is the major advantage of using these types of object detection in real time for better accuracy.

3.5 Applications of Object Detection

The major applications of object detection are Facial recognition, People counting, Self-driving cars Industrial quality check and Security.

4 Tensorflow Object Detection

Tensorflow is an open-source software library developed by the Google for numerical computation. The Tensorflow architecture is shown in figure 3.It is now widely used by many large companies. Tensorflow provides an interface for the application to execute various computations[11]. In TensorFlow [12] object detection, the computations are done with the help of dataflow graphs. The nodes in the graph represents the mathematical operation and the edges represent the data. The data are represented in the form of multi-dimensional arrays which are otherwise called as tensors as shown in figure 2.

5 Edge Computing for Quick Processing

Edge computing[13] is a distributed computing paradigm which can be easily defined as the processing of data where it is created. It emerged to reduce the bandwidth usage and time response in the system. The latency in the network can be minimized. The data processing burden in the cloud can be reduced. There are three levels of computing, the top level of computing is cloud computing, the low level of computing hierarchy is edge computing. An extended version of edge computing
is called fog computing which lies between the cloud and edge systems. Fog nodes choose where the data should be processed whether locally in data sources or in the centralized cloud servers as shown in figure 4. Major tasks of Edge computing are Data Sources, Artificial Intelligence and Actionable Insight.

6 Proposed system Implementation

The implementation of the IoT enabled security alert system consists of input acquisition which is generally a gameplay of popular Need For Speed game in First Person Camera Mode. The input can also be captured from a live camera. Once the input video is captured, it is divided into multiple image frames per second, it is then given to the Tensorflow module for object detection and collision detection of vehicles.

In Tensorflow module, a pre-trained dataset model can be used for object detection. The COCO dataset (Common Objects in Context) is mostly used because it reduces the training of custom object models. COCO dataset is a large scale object detection, segmentation and captioning dataset. It has a collection of 1.5 million object instances with 330K images being used to train the dataset. Several options of datasets are available based on system requirements and system memory. The speed and accuracy will be varied with respect to the corresponding datasets. Once the suitable model from COCO dataset is downloaded, the next step is the extraction of frozen inference graph from the downloaded model. Freezing is the process of saving a training model in a single file and making it useful for future use. So the extraction of such frozen inference graph from the model is necessary. The graph contains all the suitable checkpoint variables within the file for computations. The extracted frozen graph from the downloaded model is loaded into the memory. The label maps are then loaded into memory. For example, if 3 corresponds to a bus, then when the inference graph detects the value 3 it should display bus, for that purpose label maps are loaded. Each object contains unique labels associated with it. The images data are converted to NumPy array for processing. The dimensions of the images are expanded to large size. The position of detected objects in the image is represented in the form of boxes. The detection scores and the detection classes of corresponding boxes will be generated. This is used to identify objects with appropriate scores.

The output of the object detection is visualized in a separate window. The output window will show the objects detected with the percentage values. The occurrence can be detected if the scores of objects such as car, bus, and truck is greater than 0.5. The relative distance between these objects is calculated based on localization of objects. If the values reduces below 0.75, a warning will be displayed in the output window and a warning sound
will be generated simultaneously. The warning images will be captured and sent to the cloud instances for accuracy and to obtain the repeated accidents patterns within a targeted range. The overall information flow is shown in figure 5.

7 Results

AWS instances are created as shown in figure 5 and its created details are displayed in figure 6. Any IoT devices can act as edge computing to generate the warning messages to the drivers. More frequently accidents occurred places images or warning messages are stored in Cloud for future alert at that place. This instance is accessed in the Ubuntu terminal using SSH command as shown in figure 7. SCP protocol is used to upload the image files that are captured into the Cloud instance. Displaying the files after uploading into the instance in figure 8. Warning message generation in the web page is shown in figure 9.

7 Conclusion

The implementation is carried out virtually by using the video feeds from gameplay. However it is expected that it can work well in real time environment. As of now, while the car is in motion, if any car reaches the proximity limit the warning will be shown in the screen. On that very instant, the screen is captured and the image is stored. It is then sent to the server using SCP protocol. For future work, in addition to sending the images, the location information will also be sent to the server. To accurately detect the location, GPS in the car is used.
References

[1] K. Ashton et al., “That internet of things thing,” RFID journal, vol. 22, no. 7, pp. 97–114, 2009.
[2] Chen, S., Xu, H., Liu, D., Hu, B., & Wang, H. (2014). “A vision of IoT: Applications, challenges, and opportunities with china perspective”. IEEE Internet of Things journal, 1(4), 349-359
[3] Bharti Nathani1 , Rekha Vijayvergia2 “The Internet of Intelligent Things: An Overview” 2017 International Conference on Intelligent Communication and Computational Techniques (ICCT) Manipal University Jaipur, Dec 22-23, 2017
[4] Masumi Nakaoka1 , Pongsathorn Raksincharoensak2 and Masao Nagai3 “Study on Forward Collision Warning System Adapted to Driver Characteristics and Road Environment “ International Conference on Control, Automation and Systems 2008 Oct. 14-17, 2008 in COEX, Seoul, Korea
[5] Sushil kumar1 , Vishal Kumar2 “AUTOMATIC EMERGENCY BRAKING SYSTEM” International Journal of Research In Science & Engineering Volume: 1 Issue: 3
[6] Kang, M.-S., & Lim, Y.-C. (2017). “High performance and fast object detection in road environments”. 2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA). doi:10.1109/ipta.2017.8310148
[7] Howarth, J. W., Bakker, H. H. C., & Flemmer, R. C. (2000). Feature-based Object Recognition. 2009 4th International Conference on Autonomous Robots and Agents. doi:10.1109/icara.2000.4804015
[8] Viola-Jones object detection framework, [Online] Available: https://en.wikipedia.org/wiki/Viola-Jones_object_detection_framework
[9] Lee, S.-H., Bang, M., Jung, K.-H., & Yi, K. (2015). An efficient selection of HOG feature for SVM classification of vehicle. 2015 International Symposium on Consumer Electronics (ISCE).doi:10.1109/isce.2015.7177766
[10] Zhou, X., Gong, W., Fu, W., & Du, F. (2017). Application of deep learning in object detection. 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS).doi:10.1109/icis.2017.7960069
[12] Tensor Flow, [Online] Available: https://www.tensorflow.org/
[13] Mohammed Alrowaily “Secure Edge Computing in IoT Systems: Review and Case Studies” 2018 Third ACM/IEEE Symposium on Edge Computing