Driver Distraction Detection using Deep Learning and Computer Vision

Kusuma.S[1], Divya Udayan.J[2], Aashay Sachdeva[3]

[1]Computer Science and Engineering, Madanapalle Institute of Technology & Science, Madanapalle
[2]School of Computer Science and Engineering, Vellore Institute of Technology, Vellore
[3] School of Information Technology and Engineering, Vellore Institute of Technology, Vellore

Abstract -- **Driver distraction is a foremost cause for motor vehicle accidents and incidents. Driving requires an intensive amount of concentration otherwise the results can be fatal. Yet, most motor vehicles have no system in place to assist the driver when he is feeling drowsy, fatigued or distracted. In this research, we developed a system which detects the driver's drowsiness by using deep learning and computer vision. Whenever the driver is not concentrating, an alarm ring's. Image recognition is made possible through a model called deep convolutional neural networks (CNN). CNN can achieve good performance even on difficult image recognition tasks. Convolutional Neural Networks is a class of artificial neural networks. In this research, we used state of the art model for estimating the position of the face and the eyes to help in the better detection and reduce false positives and false negatives.**

Keywords-- Deep Learning, Convolutional Neural Network, Object detection, Computer Vision

I. INTRODUCTION

Computer vision and machine learning have created an all-new ecosystem for the tech industry. These spans from the health industry to identify diseases augmented and virtual reality to create an immersive experience, motion capture technology, and vehicle automation to name a few. These fields have changed the global tech scenario and have paved way for technologies which might have seemed impossible a few years ago. Autonomous driving is a result of advancements in computer vision and can renovate the transportation industry forever. Imagine a future where accidents can be avoided, traveling alone at any time will not be risky and there would be no traffic jams. With AR advancing, smart helmets too would become a mainstream technology soon. As indicated by the National Highway Traffic Safety Administration, one of every ten deadly crashes and two of every ten injury crashes were accounted for due to the driver distraction [1]. This research work aims to create a pipeline to detect distraction of driver using deep learning techniques for robust hand, eye and face movement detections of the driver. The aim of this work is to create a smart pipeline with which we can effectively check whether a driver is feeling distracted while he is driving.

In this research, we used convolutional Neural Network based object detection for hand, eye, and face to make the detection more precise. We proved that the CNN is able to generalize better and work better in tough conditions as compared to hear based classifiers. We will be using transfer learning to train the network 10 times faster using tensor flow object detection API.

The rest of this paper is structured as follows. Section 2 presents the overview of the literature survey. Section 3 describes the overview of the system architecture. Section 4 presents the implementation and results followed by some discussion. Finally, Section 5 concludes with some indication for future work.

II. LITERATURE SURVEY

A lot of advancement has taken place in computer vision due to deep learning. This has led to better and more robust algorithms. These algorithms have been used in the industry to solve a lot of complex computer vision problems. This also includes better key point estimation of the face.

Christopher Streiffer, created DarNet, a multi-modular information accumulation and investigation framework intended to identify and characterize distracted driving behavior. Driver diversion is distinguished as a critical supporter of motor vehicle accidents and wounds. DarNet, a combined convolutional & intermittent neural system that can break down driving picture. IMU sensor information to figure out how to recognize up to 6 classes of driving practices with expanded precision [2]. Hermannstadter applied a driver model from literature to real-road driving of a distraction experiment to study the driver's state. This model features a processing delay and a neuromuscular motor component as well as an anticipatory and a compensatory tracking component [3]. Koesdwiady A, presented a conclusion to end the profound learning answer for driver diversion acknowledgment. In the proposed structure, the highlights from pre-prepared convolutional neural systems VGG-19 are removed. In spite of the variety in enlightenment conditions, camera position, driver's ethnicity, sexual orientations in our dataset, our best-adjusted model, VGG-19 has accomplished the most noteworthy test exactness of 95% and a normal precision of 80% for every class [4].

Abouelnaga introduced the dataset for "Diverted driver" pose estimation with more diversion stances than existing choices. Likewise, we propose a dependable framework that accomplishes a 95.98% driving stance arrangement precision. The framework comprises a hereditarily weighted troupe of CNN [5]. Karahan developed an eye detection method by using deep learning method. The considered network has 3 convolution layers and 3 max-pooling layers. This model has been trained with 52K negative and 16K positive image patches. The last layer is the classification layer which operates a softmax algorithm. The trained model has been tested with images, which were provided on FDDB and CACD datasets, and also compared with eye detection algorithm [6]. Murphy focused on the advantages and disadvantages of each approach that have been published on this topic. They compared these systems by focusing on their ability to estimate coarse and fine head pose, highlighting

approaches that are well suited for unconstrained environments [7]. A method to detect and differentiate between hands in an egocentric video using robust appearance models with CNN, a proposed novel first-person dataset with the vigorous interaction between the persons along with the hand pose segmentation [8]. Hssayeni contrasted the execution of CNN's and customary highlights encouraged into a Support Vector Machine (SVM) classifier for logically identifying diverted drivers utilizing information from a dashboard camera. Three surely understood CNN models have been contrasted with customary carefully assembled features for automatically distinguishing if the drivers are taking part in diverting practices in light of pictures from a dashboard camera. ResNet-152 yielded the most accuracy of 84.2% utilizing the softmax layer as a classifier which is superior to VGG-16 precision and much superior to Alex Net [9].

III. OVERVIEW OF THE SYSTEM ARCHITECTURE

In deep learning, we learn a function f to map input X to output Y with minimal loss on the test data.

$$Y = f(X) + b \quad (1)$$

Training: Machine learns f from labeled training data

Testing: Machine predicts Y from unlabeled testing data

Once in a while f is entangled. In natural language issues, vast vocabulary sizes mean bunches of highlights. Vision issues include bunches of visual data about pixels. The learning methods we have secured so far do well when the information we are working with isn't madly intricate, however, it's not clear how they would sum up to situations like these.

Deep learning is better than average at learning f, especially in circumstances where the information is intricate. Actually, simulated neural systems are known as all-inclusive capacity approximations since they are ready to take in any capacity, regardless of how wiggly, with only a solitary hidden layer.

Hardware Components Required to build the model are camera module, GPU, Cloud processing and software components are open CV, python, NumPy, TensorFlow, Pandas, Amazon AWS EC2 and Keras.

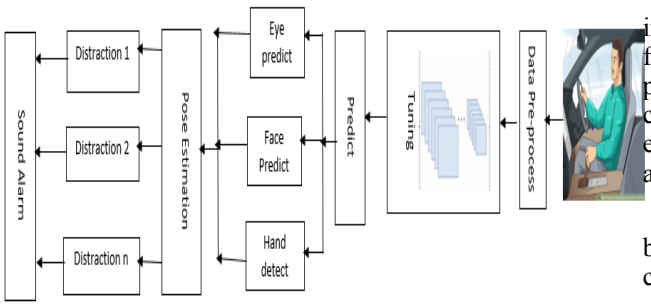

Fig.1. Deep Learning framework

A. Eye predict

This module is trained on an eye dataset using a Convolutional Neural network. It consists of three functions namely:

a) Data Preprocess to Preprocess the eye dataset before feeding it into the network for better performance and convergence.

b) Hyperparameter tuning for Fine tuning the Network. It Includes-Network initialization, Number of epochs, No of Kernels, Number of layers, Batch size.

c) Predict on the test set accuracy of the model to see how well it is generalizing. Finally Save the model at every checkpoint so as not to train it again and again.

B. Face predict

This module is trained on a face annotated dataset a Convolutional Neural Network. It consists of three functions-

a) *Data Preprocess* - Preprocess the face annotated dataset before feeding it into the network for better performance and convergence.

b) *Hyperparameter tuning* - Fine tuning the Network

c) *Predict* - Predict on the test set accuracy of the model to see how well it is generalizing.

d) *Model save* - Save the model at every checkpoint so as not to train it again and again.

e) *Get Frames*- Get frames in real-time from the camera (either usb camera or webcam). Pass it to an open CV reader to process each frame separately.

f) *Pose Estimation*- Estimate whether the driver is feeling distracted using eye predict and face predict and using the Real-Time Eye Blink Detection using Facial landmarks.

g) *Sound Alarm*- If the distraction level is not in the safe range, then an alarm is sounded to bring the driver into an attentive state again.

C. Hand detect

The hand detector model is constructed by utilizing data from the Ego Hands Dataset. This dataset works well for several reasons. It contains high quality, pixel level annotations more than where hands are located across 4800 images. All images are captured from an egocentric view (Google glass) across 48 different environments (indoor, outdoor) and activities (playing cards, chess, Jenga, solving puzzles etc.).

The hand locator display is constructed utilizing information from the Ego Hands Dataset. This dataset functions admirably for a few reasons. It contains excellent, pixel level explanations more than where hands are situated crosswise over 4800 pictures. All pictures are caught from an egocentric view across more than 50 different environments and exercises.

a) *Data Preprocess* - Preprocess the ego hand dataset before feeding it into the network for better performance and convergence.

b) *Hyperparameter tuning* - Fine tuning the Network.

c) Predict - Predict on the test set accuracy of the model to see how well it is generalizing.

d) *Model save* - Save the model at every checkpoint so as not to train it again and again.

e) *Get Frames*- Get frames in real-time from the camera (either USB camera or webcam). Pass it to an open cv reader to process each frame separately.

f) *Hand Detect*- Estimate the number of hands present in the frame.

g) *Sound Alarm*- If the number of hands is less than two for more than 3 frames, raise an alarm.

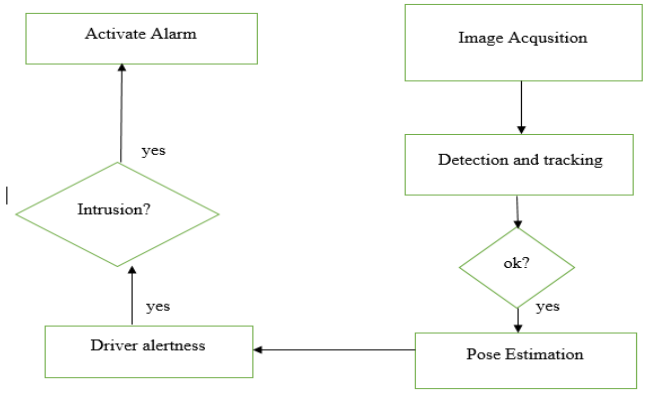

Fig.2.Detection System

The general construction of our system as presented in the fig.2. consists of five major modules: 1) Image acquisition 2) Detection and tracking 3) Pose Estimation 4) Driver alertness 5) Alarm Activation. The image acquisition is based on a micro camera placed in the vehicle searching to near Infrared Image(IR). The detection and tracking stage is responsible for separation and image processing. In the pose estimation phase, we guess some parameters from the images in order to spot some visual deeds easily obvious in people experiencing intrusion: leisurely eyelid movement, smaller gradation of eye opening, eye blink, hand predict, frequent nodding and face pose. Finally, in the driver alertness evaluation stage, we blend all individual structures obtained in the previous stage using a deep learning technique, yielding the driver alertness level. An alarm is activated if this level exceeds a certain extent [10].

IV. EXPERIMENTS

A. *Eye Detect*

Frames are passed continuously to the eye detect module and it sends back the results in real-time. The webcam is accessed using open cv Video Stream and open cv is used to make the lines using cv2 draw function as shown in fig 3. With the algorithm's robustness towards wearing specs also, which most naïve algorithms have hard time figuring out.

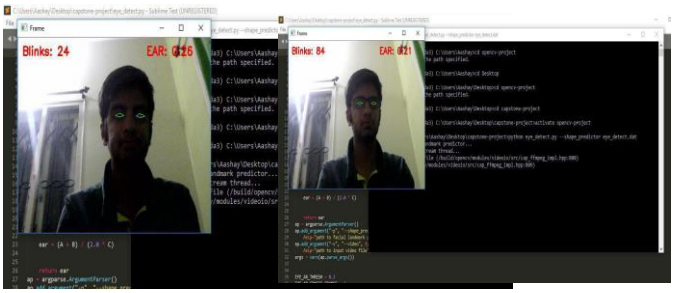

Fig.3. Eye Detect Algorithm in action.

B. *Face detect*

The facial key point detection algorithm in action is represented in the Fig.4. The network is passed features from the previous layers to learn a better representation and improve detection as after pooling the features available to the network decrease and it is not able to make precise detections

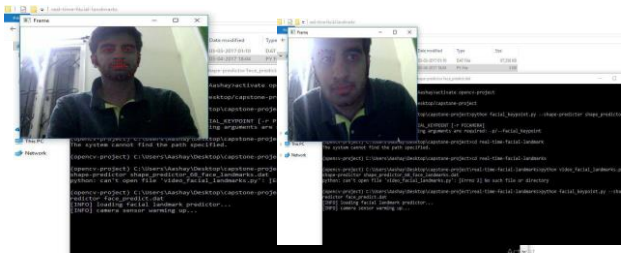

Fig.4. Facial key point Detect Algorithm in action and in different position

Fig.4 Shows the robustness of the algorithm towards the position of a face and bad lighting conditions. If for any frame the key points aren't detected, that means the person is not looking straight and an alarm is sounded.

C. *Hand detect*

Fig.5 Show the hand detection working in real-time. The Algorithm utilizes the idea of exchange learning to prepare speeder utilizing Tensor flow question recognition API.

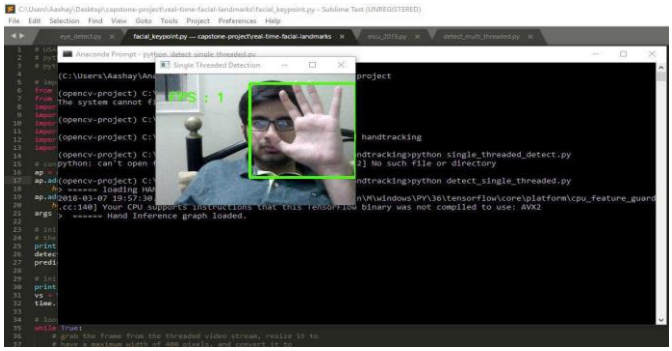

Fig.5. Hand Detect Algorithm in action.

V.RESULTS

This section presents the results on the driver distraction detection. The challenging task is to collecting the data set to evaluate the system properly for testing. The mentioned responses are obtained from each of the drivers.

Table 1. Parameters for driver distraction detection levels.

Test parameters	**Total Number of observations**	**Total Number of hits**	**Percentage of hits**
Eye Detect Normal	50	47	94.0%
Eye Detect Wearing Specs	50	45	90.0%
Eye Blink Detection	50	47	94.0%
Face Detect Normal	50	43	86.0%
Face Detect Bad Light Condition	50	41	82.0%
Hand Detect	50	45	90.0%

V.CONCLUSION AND FUTURE SCOPE

The research work shows how vehicles can be made even safer using state of the art algorithms and reduce false negatives. This software can be integrated easily into existing systems and massively decrease the number of accidents that happen on the road. While we have achieved state of the art results, the hardware right now present is not good enough to run these algorithms efficiently. For example, the hand detection only gives 1 fps while we need at least 40-50 fps to make the algorithm effective in real life settings. Our eyes process the surroundings at 60 fps. The hardware required during the training period is pretty expensive. It costed me to use $600 of student credits of google cloud platform and Amazon web services (AWS) to optimize the network. As the research community gets better at training deep neural network, these costs will reduce. The algorithms cannot run on standalone computer devices like raspberry pi which makes them a hardware & software entity as difficult.

We can use memory efficient algorithms which can work on simple computers. Well, deep learning techniques could be used to reduce the training time. Using auto encoding of features to automate the preprocessing step of data could have been a better option. The algorithm could be served as an API so that computation happens on the cloud. We should use containerized services like kubernetes and docker to ship them as a single entity.

REFERENCES

[1] National Center for Statistics and Analysis, Distracted Driving 2014 (Traffic Safety Facts Research Note). No. DOT HS 812 260, National Highway Traffic Safety Administration (NHTSA), April 2016.

[2] Christopher Streiffer DarNet: A Deep Learning Solution for Distracted Driving Detection, 2017 ACM. 978-1-4503-5200-0/17/12DOI: 10.1145/3154448.3154452.

[3] Hermannstadter P & Yang, B. (2013, October). Driver distraction assessment using driver modeling. In Systems, Man, Cybernetics (SMC), 2013 IEEE International Conference on (pp. 3693-3698). IEEE.

[4] Koesdwiady A., Bedawi S.M., Ou C., Karray F. (2017) End-to-End Deep Learning for Driver Distraction Recognition. In: Karray F., Campilho A., Cheriet F. (eds) Image Analysis and Recognition. ICIAR 2017. Lecture Notes in Computer Science, vol 10317. Springer, Cham.

[5] Abouelnaga, Y., Eraqi, H. M., & Moustafa, M. N. (2017). Real - time Distracted Driver Posture Classification. arXiv preprint arXiv:1706.09498.

[6] Karahan, Ş., & Akgül, Y. S. (2016, May). Eye detection by using deeplearning. In Signal Processing and Communication Application Conference (SIU), 2016 24th (pp. 2145-2148). IEEE.

[7] Murphy-Chutorian, E., & Trivedi, M. M. (2009). Head pose estimation in computer vision: A survey. IEEE transactions on pattern analysis and machine intelligence, 31(4), 607-626.

[8] Bambach, S., Lee, S., Crandall, D. J., and Yu, C. 2015. "Lending A Hand: Detecting Hands and Recognizing Activities in Complex Egocentric Interactions," in ICCV, pp. 1949–1957

[9] Hssayeni, Murtadha D; Saxena, Sagar; Ptucha, Raymond; Savakis, Andreas.(2017) "Distracted Driver Detection: Deep Learning vs HandcraftedFeatures"D**OI:**https://doi.org/10.2352/I1173.2017.10.IAWM-162.

[10] Luis M. Bergasa, Associate Member, IEEE, Jesús Nuevo,Miguel A. Sotelo, Member, IEEE, Rafael Barea, and María Elena Lopez," Real-Time System for Monitoring Driver Vigilance", IEEE Transactions On Intelligent Transportation Systems,Vol. 7, No. 1, March 2006.