Vehicle driver's distraction using Convolution Neural Networks

Shivani Gupta
Indiana University Bloomington
shivgupt@umail.iu.edu
(646)312-9159

Snehil Vishwakarma
Indiana University Bloomington
snehvish@umail.iu.edu
(862)215-0108

ABSTRACT

According to the CDC motor vehicle safety division, one in five car accidents is caused by a driver using mobile phones. Every year, the count of car accidents by the driver using mobile phones increases. The rest includes a big number of other frequently occurring prohibited actions like drinking, eating, smoking, and even reaching out to the back. Using mobile phone by the driver is the majorly occurring distraction to a driver leading to road accidents, increases the probability of accidents due to driver's distraction. In this project we are developing a system to detect 7 different distractions in a huge dataset of driver's moments. The possible distractions are: texting, talking on the phone, operating the radio, drinking, reaching behind, hair and makeup, and talking to the passenger. We are using Convolutional Neural Networks to train our dataset and test it on images, given in the same dataset.

Keywords

Computer Vision; CNN; Caffe; AlexNet

1. INTRODUCTION

For improved traffic safety it is important to develop intelligent system for driver assistance that can continuously monitor the state of the driver of the vehicle and will able the insurance companies to better insure the drivers. In this project we aim to detect driver's distraction from an image taken from the passenger seat of the vehicle.

Many of the previous vision based approaches to solve this problem have used hand-crafted feature extraction and then learn classifiers on those feature, but have not showed convincing results. In this project we first followed the approach from [1] andd Trained on statefarm dataset. To improve the accuracy we then used the weights from pretrained model - AlexNet and finetuned on our trainset.

2. BACKGROUND AND RELATED WORK

Developing tools using the continuous technological advancement to reduce road and driving-related accidents are proactively being done by both the government and the automobile industry. Previous approaches use various vehicle-orientation (vehicle behavior including steering wheel movement, pressure on the accelerate, break and clutch pedals, vehicle's speed and road position, and so on) and driver-physiological (brain and heart activity, temperature, muscle movement) analysis which were highly dependent on the subject and it's personal characteristics. A lot can be bargained on, them being more specific, but due to their low accuracy on generalization of the subjects, their implementation on real-life incidents isn't productive. One more drawback to both these techniques are the high amount of hardware cost and also it's implementation in real-life driving.

The next development to this problem came with vision-based techniques. The approaches were spread out to determine different parts of the human body involved during driving and analyze those features together to determine the behavior of driver and linking it to the reaction. This solution was over-complicated and didn't give good results due to a lot of specification of the factors involved in the analysis. For these approaches to give good results, again, one had to give appropriate ideal conditions, which is one of the major drawback to real-life appliactions of these techniques.

In recent years, the application of deep learning models have enormously increased in vision-based tasks. The representation of problem and it's learning in the form of abstract features is the crux of deep learning techniques. For our stated problem, till date the most successful deep machine learning model is the Convolutional Neural Network (CNN). This deep learning model is a multi-layer hierarchical structure of various types of operations (trainable filters and local regional pooling operators, including image parts transformations) bind together in an iterative and back-propagated neural network model. Our project aims to implement and analyze the application of self-trained and fine-tuning of pre-trained CNN model.

3. METHODOLOGY

In this paper, we first apply Convolutional Neural Network architecture to to recognize driving postures and automatically learn features with minimum domain knowledge from the raw input images.

3.1 Architecture

The network consists of five convolution stages followed by three fully connected layers. Each convolution stage includes

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

a convolutional layer, a rectifier linear unit activation layer, local response normalization layers, and a max-pooling layer. The final layer is connected to a softmax layer. The structure of the networks and the hyper-parameters were initialized based on previous work.

3.1.1 Layers

ImageData: holds raw pixel values of the image which is re-sized to 227x227.

CONV: computes the output of neurons that are connected to local regions in the input, each computing a dot product between their weights and their respective local regions connections. The gaussian filter of kernel size 11 is used. The operations have common weight vectors that are initialized as random and will learn to detect specific patterns or features.

RELU: rectifier linear unit (ReLU) activation function layer is used to form a non-linear complex model. It transforms the input value non-linearly to the output value of the neuron. Relu is used for both convolution as well as fully connected layer.

POOL: In order to down sample the spatial dimensions while preserving only the most critical information and also to avoid detected features to be sensitive to the precise positions of the input pattern, a max pooling approach is used. This layer summarizes the output of multiple neurons from convolutional layers.

3.2 Features

In this project we first used a deep CNN in which trainable filters and max pooling operations are applied alternatively to automatically explore features and visualized the features learned at each convolution layer.

3.3 Pre-Trained Model

The accuracy obtained in the previous model were not very satisfactory, thus the initial approach was later modified to fine-tune the pre-trained model of AlexNet, trained on ImageNet data-set, in order to improve accuracy.

4. RESULTS

The accuracy of finetuned AlexNet model was observed 99.43% when tested on the test images from similar dataset. It was observed that out of 19 images that constitute the error were actually labelled wrong in the dataset. As seen figure 1 and 2 below the predicted label is much suited to the image as opposed to true label.

To check the robustness of the classifier a couple of existing datasets were modified (different parts of the test image were covered with a black patch), and the resulting accuracy were still above 85%. The relevant information can be seen in Figure 3 and Figure 4.

The same model was then used to test images from a smaller hand labeled dataset of random drivers that were not present in the train set and the accuracy dropped to 80%.

5. CONCLUSIONS

The accuracy of the model increased from 56% to 99.43% when weights from pretrained AlexNet model were used. By testing the images covered with patches it was observed the features learned accurate. Images that in which irrelevant

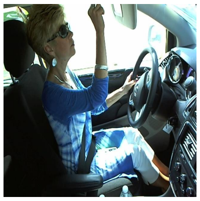

Figure 1: True Label: talking to passenger; Predicted Label: hair and makeup

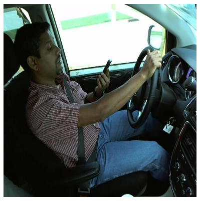

Figure 2: True Label: talking on the phone - left; Predicted Label: texting - left

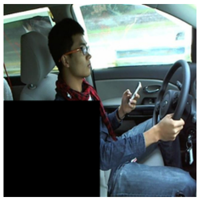

Figure 3: True Label: texting - left; Predicted Label: texting - left

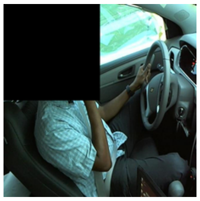

Figure 4: True Label: hair and makeup; Predicted Label: talking on the phone - right

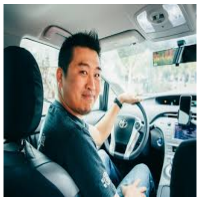

Figure 5: True Label: reaching behind; Predicted Label: safe driving

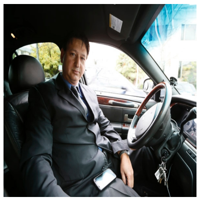

Figure 6: True Label: talking to passenger; Predicted Label: talking to passenger

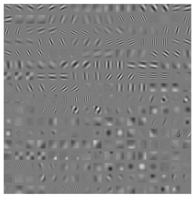

Figure 7: Confusion Matrix of State Farm test set

portion was covered, eg. Figure 3, were classified correctly even after adding a patch. Whereas Figure 4 shows the image which was classified wrong because a crucial part of the image was hidden. This image when tested without patch was classified correctly.

Also, it was observed that the dataset was over-fitted to the angle at which the images were taken and maybe to the 10 drivers that constituted the train-set. Since the original test set and train set have similar drivers the accuracy is very high but it dropped drastically when new driver images are tested.

To reduce the over-fitting problem model should be fine-tuned on a much more diverse dataset.

5.1 References

1. Yan, Chao; Coenen, Frans; Zhang, Bailing: 'Driving posture recognition by convolutional neural networks', IET Computer Vision, 2016, 10, (2), p. 103-114, DOI: 10.1049/iet-cvi.2015.0175 IET Digital Library, http://digital-library.theiet.org/content/journals/10.1049/iet-cvi.2015.0175
2. https://www.kaggle.com/c/state-farm-distracted-driver-detection/data
3. https://kb.iu.edu/d/bcqt
4. caffe.berkeleyvision.org