Real-time Distracted Driver Posture Classification

Yehya Abouelnaga, Hesham M. Eraqi, and Mohamed N. Moustafa
Department of Computer Science and Engineering, School of Sciences and Engineering
The American University in Cairo, New Cairo 11835, Egypt
{devyhia,heraqi,m.moustafa}@aucegypt.edu

Abstract: Distracted driving is a worldwide problem leading to an astoundingly increasing number of accidents and deaths. Existing work is concerned with a very small set of distractions (mostly– cell phone usage). Also, for the most part, it uses unreliable *ad-hoc* methods to detect those distractions. In this paper, we present the first publicly available dataset for "distracted driver" posture estimation with more distraction postures than existing alternatives. In addition, we propose a reliable system that achieves a 95.98% driving posture classification accuracy. The system consists of a genetically-weighted ensemble of Convolutional Neural Networks (CNNs). We show that a weighted ensemble of classifiers using a genetic algorithm yields in better classification confidence. We also study the effect of different visual elements (i.e. hands and face) in distraction detection by means of face and hand localizations. Finally, we present a thinned version of our ensemble that could achieve a 94.29% classification accuracy and operate in a real-time environment.

1 Introduction

The number of road accidents due to distracted driving is steadily increasing. According to the National Highway Traffic Safety Administration (NHTSA), in 2015, 3,477 people were killed, and 391,000 were injured in motor vehicle crashes involving distracted drivers [21]. The major cause of these accidents was the use of mobile phones. The NHTSA defines distracted driving as "any activity that diverts attention from driving", including: a) talking or texting on one's phone, b) eating and drinking, c) talking to passengers, or d) fiddling with the stereo, entertainment, or navigation system [21]. The Center for Disease Control and Prevention (CDC) provides a broader definition of distracted driving by taking into account visual (i.e. taking one's eyes off the road), manual (i.e. taking one's hands off the driving wheel) and cognitive (i.e. taking one's mind off driving) causes [23]. We believe that the detection of distracted driver's postures is key to further preventive measures. We envision a future where smart vehicles could detect such distraction, and warn the driver against it (i.e. by beeping). That could also help law enforcement to identify distraction on highway using radar cameras, and penalize certain forms of distraction.

The 2015 Global Status Report of the World Health Organization (WHO) reported an estimated 1.25 million deaths yearly due to road traffic accidents worldwide [19]. With approximately 89% of accidents resulting from human errors, self-driving cars will play a vital role to significantly reduce this number and ultimately save human lives. In fact, latest commercial self-driving cars still require drivers to pay attention and be ready to take back control of the wheel [8]. This is what makes distracted drivers detection an important system component in self-driving cars.

Research in the field of distracted driving detection follows the definitions presented in [21] and [23]. It detects manual, visual, or cognitive types of distractions. Cognitive distractions deal with tasks of listening, conversing, daydreaming, or just becoming lost in thought. In this form of distraction, the driver is "mentally" distracted from safe driving even though they are in a safe driving posture. Visual distractions often refer to situations where the driver takes their eyes off the road due to either "the presence of salient visual information away from the road causing spontaneous off-road eye glances and momentary rotation of the head" or the use of multimedia devices (i.e. cell phones, navigation or entertainment systems) [10]. Visual distractions are coined in the following terms: "sleepiness", "drowsiness", "fatigue", and "inattention". And, they usually depend on facial landmarks detection and tracking. Manual distractions are mainly concerned with driver's activities

1st Conference on Robot Learning (CoRL 2017), Mountain View, United States.

other than safe driving (i.e. reaching behind, adjusting hair and makeup, or eating and drinking). In this kind of distraction, authors often tend to depend heavily on hand tracking and driving posture estimation. In this paper, we focus only on "manual" distractions where a driver is distracted by texting or using cell phone, calling, eating or drinking, reaching behind, fiddling with the radio, or adjusting hair and makeup.

We present a real-time distracted driver posture estimation system using a weighted ensemble of convolutional neural networks, a challenging distracted driver's dataset on which we evaluate our proposed solution, and an annotation tool [1] for action labeling in those videos.

2 Literature Review

The work in the distracted driver detection field over the past seven years could be clustered into four groups: multiple independent cell-phone usage detection publications, Laboratory of Intelligent and Safe Automobiles in University of California San Diego (UCSD) datasets and publications, Southeast University Distracted Driver dataset and affiliated publications, and recently, StateFarm's Distracted Driver Kaggle competition.

2.1 Cell Phone Usage Detection

[4] presents an SVM-based model that detects the use of mobile phone while driving (i.e. distracted driving). Their dataset consists of frontal image view of a driver's face. They also make pre-made assumptions about hand and face locations in the picture. [2] presents another SVM-based classification method to detect cell phone usage. However, their dataset is collected from transportation imaging cameras that are deployed in highways and traffic lights which is, indeed, more competitive. [6] uses AdaBoost classifier and Hidden Markov Models to classify a Kinect's RGB-D data. Their solution depends on indoor-produced data. They sit on a chair and a mimmic a certain distraction (i.e. talking on the phone). This setup misses two essential points: the lighting conditions and the distance between a Kinect and the driver. In real-life applications, a driver is exposed to a variety of lighting conditions (i.e. sunlight and shadow). [31] suggests using a Hidden Conditional Random Fields (HCRF) model to detect cell phone usage. Their model operates face, mouth, and hand features of images obtained from a camera mounted above the dashboard. [11] devised a Faster-RCNN model to detect driver's cell-phone usage and "hands on the wheel". Their model is mainly geared towards face/hand segmentation. They train their Faster-RCNN on the dataset proposed in [7] (that we also use in this paper). Their proposed solution runs at a 0.06, and 0.09 frames per second for cell-phone usage, and "hands on the wheel" detection. [24] tackles the problem of cell phone usage detection. Their approach doesn't hold any static assumptions though (i.e. in which region of the image a face is expected to be found). They use a Supervised Descent Method (SDM) to localize the face landmarks, and then, extract two bounding boxes to the left and the right side of the face. They train a classifier on each of the two regions to detect cell phone usage: right hand, left hand, or no usage. Using a histogram of gradients (HOG) and an AdaBoost classifier, they achieve a 93.9% classification accuracy and operate in a near real-time speed (7.5 frames per second).

2.2 UCSD's Laboratory of Intelligent and Safe Automobiles Work

[15] presents an vision-based analysis framework that recognizes in-vehicle activities using two kinect cameras that provide frontal and back views of the driver. Their approach provides "hands on the wheel" information (i.e. left hand only, both hands, no hands), and uses these information to detect three types of distractions: adjusting the radio, operating the gear, and adjusting the mirrors. [16] presents a fusion of classifiers where the image is to be segmented into three regions: wheel, gear, and instrument panel (i.e. radio). It proposes a classifier for each segment to detect existence of hands in those regions. The hand information (i.e. output of the classifiers) is passed to an "activity classifier" that infers the actual activity (i.e. adjusting the radio, operating the gear). [18] extends existing research to include eye cues to previously existing head and hands cues. However, it still considers three types of distractions: "wheel region interaction with two hands on the wheel, gear region activity, and instrument cluster region activity". [17] presents a region-based classification approach. It detects hands presence in certain pre-defined regions in an image. A model is learned for each region separately. All regions are later joined using a second-stage classifier.

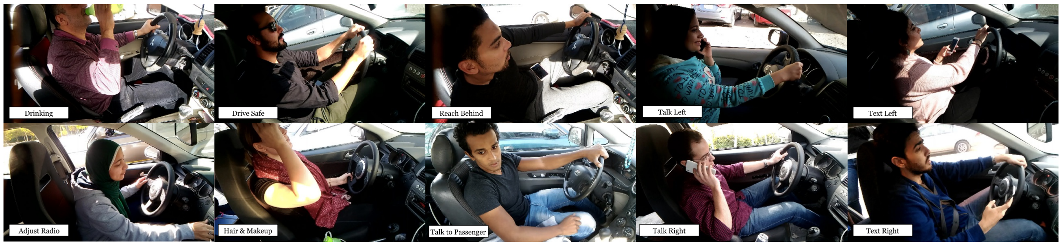

Figure 1: Examples of the proposed Distracted Driver's Dataset. In a column-level order, postures are: drinking, adjusting the radio, driving in a safe posture, fiddling with hair or makeup, reaching behind, talking to passengers, talk on cell phone using left hand, talk on cell phone using right hand, texting using left hand, and texting using right hand.

2.3 Southeast University Distracted Driver Dataset

[34] designs a more inclusive distracted driving dataset with a side view of the driver and more activities: grasping the steering wheel, operating the shift lever, eating a cake and talking on a cellular phone. It introduces a contourlet transform for feature extraction, and then, evaluates the performance of different classifiers: Random Forests (RF), k-Nearest Neighbors (KNN), and Multi-Layer Perceptron (MLP). The random forests achieved the highest classification accuracy of 90.5%. [33] showed that using a multiwavelet transform improves the accuracy of multilayer perceptron classifier to 90.61% (previously 37.06% in [34]). [32] showed that using a Support Vector Machine (SVM) with an intersection kernel, followed by Radial Basis Function (RBF) kernel, achieved the highest accuracies of 92.81% and 94.25%, respectively (in comarison with [34] and [33]). After testing against other classification methods, they concluded that an SVM with intersection kernel offers the best real time quality (67 frames per second) and better classification performance. [35] improves the Multilayer Perceptron classifier using combined features of Pyramid Histogram of Oriented Gradients (PHOG) and spatial scale feature extractors. Their multilayer perceptron achieves a 94.75% classification accuracy. [28] utilizes Motion History Images (HMI) to make use of the data's temporality. Pyramid Histogram of Gradients (PHOG) is applied to the motion history images. A random forrest trains on the extracted features and yields a 96.56% accuracy. [29] presents a convolutional neural network solution that achieves a 99.78% classification accuracy. They train their network in a 2-step process. First, they use pre-trained sparse filters as the parameters of the first convolutional layer. Second, they fine-tune the network on the actuall dataset. Their accuracy is measured against the 4-classes of the Southeast dataset.

2.4 StateFarm's Dataset

StateFarm's Distracted Driver Detection competition on Kaggle was the first publicly available dataset for posture classification. In the competition, StateFarm defined ten postures to be detected: safe driving, texting using right hand, talking on the phone using right hand, texting using left hand, talking on the phone using left hand, operating the radio, drinking, reaching behind, doing hair and makeup, and talking to passenger. Our work, in this paper, is mainly inspired by StateFarm's Distracted Driver's competition. While the usage of StateFarm's dataset is limited to the purposes of the competition [25], we designed a similar dataset that follows the same postures.

3 Dataset Design

Creating a new dataset was essential to the completion of this work. The available alternatives to our dataset are: StateFarm and Southeast University (SEU) datasets. StateFarm's dataset is to be used for their Kaggle competition purposes only (as per their regulations) [25]. As for Southeast University (SEU) dataset, it presents only four distraction postures. And, after multiple attempts to obtain it, we figured out that the authors do not make it publicly available. All the papers ([30, 29, 28, 35, 33, 32, 34]) that benchmarked against the dataset are affiliated with either Southeast University, Xian Jiaotong-Liverpool University, or Liverpool University, and they have at least one shared author. With that being said, the collected "distracted driver" dataset is the first publicly available for driving posture estimation research.

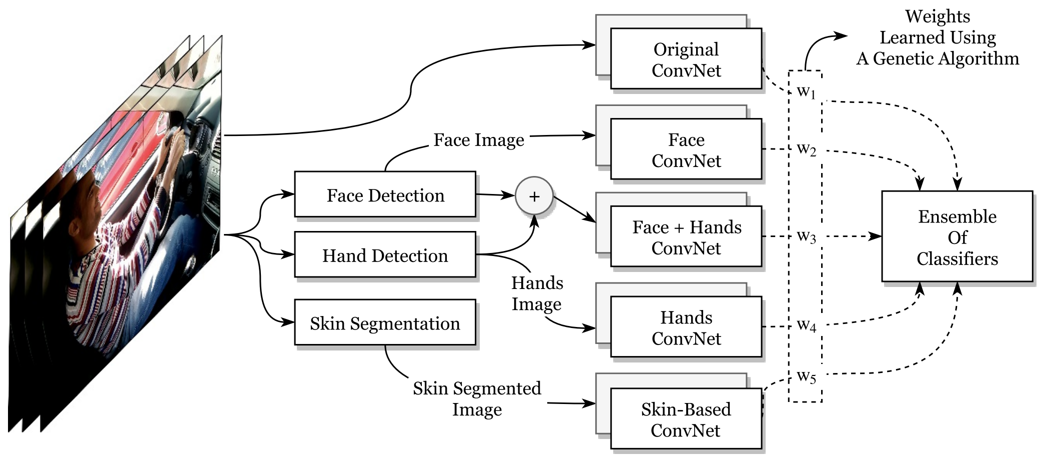

Figure 2: An overview of our proposed solution. A face detector, a hand detector, and a skin segmentor are run against each frame. For each output image (i.e. Skin, Face, Hands), an AlexNet and an InceptionV3 networks are trained (i.e. resulting in 10 neural networks: 5 AlexNet and 5 InceptionV3). The overall class distribution is determined by the weighted sum of all softmax layers. The weights are learned using a genetic algorithm.

3.1 Camera Setup

The Distracted Driver's dataset is collected using an ASUS ZenPhone (Model Z00UD) rear camera. The input was collected in a video format, and then, cut into individual images, 1080×1920 each. The phone was fixed using an arm strap to the car roof handle on top of the front passenger's seat. In our use case, this setup proved to be very flexible as we needed to collect data in different vehicles.

3.2 Labeling

In order to label the collected videos, we designed a simple multi-platform action annotation tool using modern web technologies: Electron, AngularJS, and Javascript. The annotation tool is open-source and publicly available at [1].

3.3 Statistics

We had 31 participants from 7 different countries: Egypt (24), Germany (2), USA (1), Canada (1), Uganda (1), Palestine (1), and Morocco (1). Out of all participants, 22 were males and 9 were females. Videos were shot in 4 different cars: Proton Gen2 (26), Mitsubishi Lancer (2), Nissan Sunny (2), and KIA Carens (1). We extracted 17,308 frames distributed over the following classes: Drive Safe (3,686), Talk Passenger (2,570), Text Right (1,974), Drink (1,612), Talk Left (1,361), Text Left (1,301), Talk Right (1,223), Adjust Radio (1,220), Hair & Makeup (1,202), and Reach Behind (1,159).

4 Proposed Method

Our proposed solution consists of a genetically-weighted ensemble of convolutional neural networks. The convolutional neural networks train on raw images, skin-segmented images, face images, hands images, and "face+hands" images. We train an AlexNet [13] and an InceptionV3 [26] on those five images sources. In the InceptionV3 network, we fine-tune a pre-trained ImageNet model (i.e. transfer learning). Then, we evaluate a weighted sum of all networks' outputs yielding the final class distribution. The weights are evaluated using a genetic algorithm.

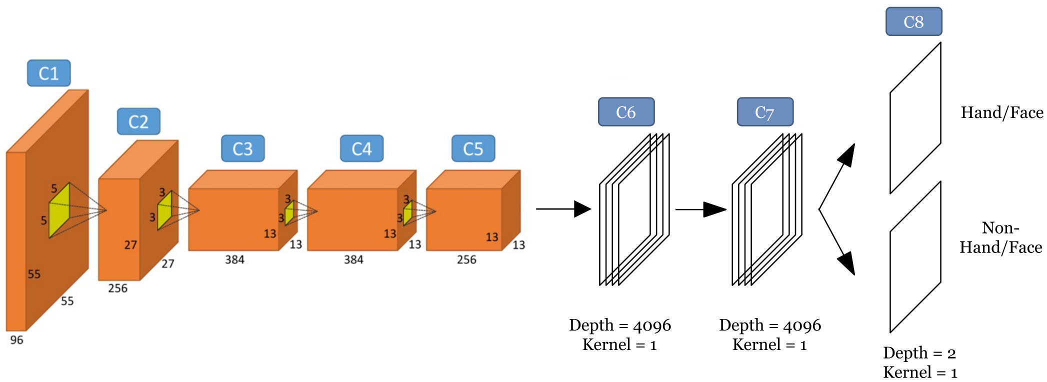

Figure 3: A modified version of AlexNet for hands/face localization. An AlexNet is trained on Object/Background images. Fully connected layers are replaced with convolutional layers with the same weights of the fully connected layers.

4.1 Skin Segmentation

Skin segmentation was a challenging problem to solve due to the different lighting conditions during driving. We use a Multivariate Gaussian Naive Bayes classifier to develop a pixel-wise skin segmentation model that operates on an HSV colorspace. Our model is very similar to [20] except that we don't use a histogram. Instead, we use a normal distribution.

$$\begin{aligned}
\mathbb{P}(x \mid skin) &\sim \mathcal{N}(\mu_{\text{Skin}}, \Sigma_{\text{Skin}}) \\
\mathbb{P}(skin|x) &= \frac{\mathbb{P}(x|skin) \cdot \mathbb{P}(skin)}{\mathbb{P}(x)} \\
\text{Model}(x) &= \frac{\mathbb{P}(skin \mid x)}{\mathbb{P}(skin \mid x) + \mathbb{P}(non\text{-}skin \mid x)} \\
&= \frac{\mathbb{P}(x \mid skin)}{\mathbb{P}(x \mid skin) + \mathbb{P}(x \mid non\text{-}skin)}
\end{aligned}$$

Note that $\mathbb{P}(skin) = \mathbb{P}(non\text{-}skin) = 0.5$ (i.e. we don't make any assumptions about existence of skin pixels in the image).

We trained our model using the UCI Skin Segmentation dataset [5]. μ_{skin}, Σ_{skin}, $\mu_{non-skin}$, and $\Sigma_{non-skin}$ are evaluated from labeled skin/non-skin values in [5]. Now, we have a normal distribution for both skin and non-skin pixels ready for use. In order to evaluate the model (i.e. skin-segment images), for each pixel, x, in the input image, we transform the RGB values to HSV, and then, feed them to our model. Then, we get a probability heat map of skin in the image. We classify a pixel to a "skin" when $\text{Model}(x) > 0.5$. Then, we cluster the skin pixels into objects and remove those with a small number of pixels. In other words, we don't expect neither faces nor hands to have a very small number of pixels.

4.2 Face & Hands Detection

We trained the model presented in [14] on the Annotated Facial Landmarks in the Wild (AFLW) face dataset [12]. The trained model achieved decent results. However, it was sensitive to distance from the camera (i.e. faces that were close to the camera were not easily detected). We found that the pre-trained model (presented in [9]) produced better results on our dataset. Given that we did not have any hand labelled face bounding boxes, we couldn't formally compare the two models. However, when randomly selecting images from different classes, we found that [9] was closer to what we expected.

As for hands detection, we used the pre-trained model presented in [3] with slight modifications. Their trained model was a binary class AlexNet that classifies hands/non-hands for different proposal windows. We transferred the weights of the fully connected layers (i.e. fc6, fc7 and fc8) into convolutional layers such that each neuron in the fully connected layer was transferred into a depth layer with a 1-pixel kernel size. This architecture accepts variant size inputs and produces variant-size outputs. The last convolutional layer has a depth of 2 (i.e. the binary classes) where

$$\text{Conv8}_{x,y,0} + \text{Conv8}_{x,y,1} = 1$$

for all x and y such that $0 \leq x < W$ and $0 \leq y < H$ where W and H are the output's width and height, respectively.

4.3 Convolutional Neural Network

For distracted driver posture classification, we trained two classes of neural networks: AlexNet and InceptionV3. Each network is trained on 5 different image sources (i.e. raw, skin, face, hands and face+hands images) yielding in 5 models per net and a total of 10 models.

We trained our AlexNet models from scratch. We didn't use a pre-trained model. As for InceptionV3, we performed transfer learning. We fine-tuned a pre-trained model [27] on the distraction postures. We removed the "logits" fully connected layer, and replaced it with 10-neuron fully connected layer (i.e. corresponding to 10 driving postures). For all of our models, we used a gradient descent optimizer with an initial learning rate of 10^{-2}. The learning rate decays linearly in each epoch with a step of

$$\frac{10^{-2} - 10^{-4}}{\text{Epochs}}$$

We trained the networks for 30 epochs. In each, we divide the training dataset into mini-batches of 50 images each.

4.4 GA-based Ensemble of Classifiers

Each classifier produces a class probability vector (i.e. output of "softmax" layer), C_1 ... C_N, such that C_i has 10 probabilities (i.e. 10 classes) and N is the number of classifiers ($N = 10$ in our situation). In a majority voting system, it is assumed that all experts (i.e. classifiers) can equally contribute to a better decision by taking the unweighted sum of all classifier outputs.

$$C_{\text{Majority}} = \frac{1}{N} \sum_{i}^{N} C_i$$

However, that is not usually a valid assumption. In a weighted voting system, we assume that classifiers do not contribute equally to the ensemble and that some classifiers might yield higher accuracy than others. Therefore, we need to estimate the weights of each classifier's contribution to the ensemble. [22] presents a variety of methods to estimate the weights. We opted to use a genetic algorithm (i.e. a search-based method).

$$C_{\text{Weighted}} = \frac{1}{\sum_{i}^{N} w_i} \sum_{i}^{N} w_i \cdot C_i$$

In our genetic algorithm, a chromosome consists of N genes that correspond to the weights w_1 ... w_N. Our fitness function evaluates the Negative Log Likelihood (NLL) loss over a 50% random sample of the population. This helps prevent overfitting. Our population consists of 50 individual. In each iteration, we retain the top 20% of the population and use them as parents. Then, we randomly select 10% of the remaining 80% of the population as parents. In other words, we have 30% of the population as parents. Now, we randomly mutate 5% of the selected parents. Finally, we cross-over random pairs of the parents to produce children until we have a full population (i.e. with 50 individuals). We ran the above procedure for only 5 iterations in order to avoid over-fitting. We selected the chromosome with the highest fitness score (test against all data points– not 50%).

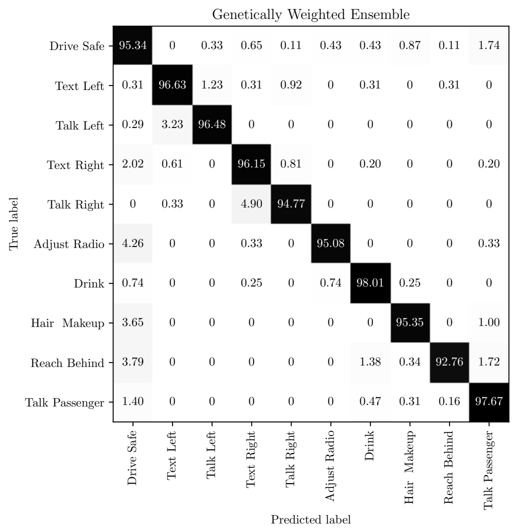

Figure 4: Confusion Matrix of Genetically Weighted Ensemble of Classifiers

5 Experiments

We divided our dataset into 75% training and 25% held out test data. Then, we ran the skin segmentation, face and hand detectors on the entire dataset. We tested all of the networks against our test dataset and obtained the results in Table 1. We notice that both AlexNet and InceptionV3 achieve best accuracies when trained on the original images. However, the accuracy doesn't majorly change in both architectures when switching from the original images to skin segmented images. Hands seem to have more weight in posture recognition than the face. "Face + Hands" images produce slightly lower accuracy than the hands images, yet, still higher than the face images. That happens due to face/hand detector failures. For example, if a hand is not found, we pass a face image to a "face + hands" classifier. This doesn't happen in individual cases of hand-only or face-only classifier because if the hand/face detection fails, we pass the original image to the hand/face classifier as a fallback mechanism. With better hand/face detectors, the "face+hands" networks are expected to produce higher accuracies than the "hands" networks.

5.1 Benchmarking

We trained and tested our models using an EVGA GeForce GTX TITAN X 12GB GPU, Intel(R) Core(TM) i7-5960X CPU @ 3.00GHz, and a 48 GM RAM. On average, AlexNet processed 182 frames per second using a GPU and 52 frames per second using a CPU. InceptionV3 processes 72 frames per second using a GPU and 5.5 frames per second using a CPU.

5.2 Real System

An ensemble of two AlexNet models (Original and Skin-segmented networks) produce a satisfactory classification accuracy (i.e. 94.29%). Meanwhile, it still maintains a real-time performance on a CPU-based system.

5.3 Analysis

In Figure 4, we notice that the most confusing posture is the "safe driving". This is due to the lack of temporal context in static images. In a static image, a driver would appear in a "safe driving" posture. However, contextually, he/she was distracted by doing some other activity. "Text Left" is mostly confused for "Talk Left" and vice versa. Same applies to "Text Right" and "Talk Right". "Adjust Radio" is mainly confused for a "safe driving" posture. That is due to lack of the previously mentioned temporal context. Apart from safe driving, "Hair & Makeup" is confused for talking to passenger. That is because, in most cases, when drivers did their hair/makeup on the left side of

Table 1: Distracted Driver Posture Classification Results

Model	Source	Loss (NLL)	Accuracy (%)
AlexNet	**Original**	**0.3909**	**93.65**
	Skin Segmented	0.3446	93.60
	Face	1.0516	84.28
	Hands	0.6186	89.52
	Face + Hands	0.8298	86.68
InceptionV3	**Original**	**0.2654**	**95.17**
	Skin Segmented	0.2937	94.57
	Face	0.6096	88.82
	Hands	0.4546	91.62
	Face + Hands	0.4495	90.88
Real-time System		0.2727	94.29
Majority Voting Ensemble		**0.1661**	**95.77**
GA-Weighted Ensemble		**0.1575**	**95.98**

their face, they needed to tilt their face slightly right (while looking at the frontal mirror). Thus, the network thought the person was talking to passenger. “Reach Behind” was confused for both talking to passenger and drinking. That makes sense as people tend to naturally look towards the camera while reaching behind. As for the drinking confusion, it is due to right-arm movement from the steering wheel to the back seat. A still image in the middle of that move could be easily mistaken for a drinking posture. “Drink” and “Talk to Passenger” postures were not easily confused with other postures as 98% and 97.67% of their images were correctly classified.

6 Conclusion

Distracted driving is a major problem leading to a striking number of accidents worldwide. Besides, its detection is an important system component in self-driving cars. In this paper, we presented a robust vision-based system that recognizes distracted driving postures. We collected a distracted driver dataset that we used to develop and test our system. Our best model utilizes a genetically weighted ensemble of convolutional neural networks to achieve a 95.98% classification accuracy. We also showed that a simpler model (only using AlexNet) could operate in real-time and still maintain a satisfactory classification accuracy. Face and hands detection is proved to improve classification accuracy in our ensemble. However, in a real-time setting, their performance overhead is much higher than their contribution.

In a future work, we need to devise a better face and hands detector. We would need to manually label hand and face proposals and use them to train a Fast RCNN to localize both faces and hands in one shot. We would love to evaluate that against our existing CNN-based localization system. In order to overcome the “safe driving” posture confusion with other classes, we would need to incorporate temporality in our decision. We shall test the performance of a Recurrent Neural Network (RNN) against sequential stream of frames. We envision a performance improvement due to temporal features.

References

[1] Y. Abouelnaga. Action Annotation Tool, 2017. URL `https://github.com/devyhia/action-annotation`.

[2] Y. Artan, O. Bulan, R. P. Loce, P. Paul, P. Rd, and W. Ny. Driver Cell Phone Usage Detection From HOV / HOT NIR Images. 5474, 2009.

[3] S. Bambach, S. Lee, D. J. Crandall, and C. Yu. Lending A Hand: Detecting Hands and Recognizing Activities in Complex Egocentric Interactions. In *The IEEE International Conference on Computer Vision (ICCV)*, dec 2015.

[4] R. Berri and A. G. Silva. A Pattern Recognition System for Detecting Use of Mobile Phones While Driving. (August), 2014. doi:10.5220/0004684504110418.

[5] R. Bhatt and A. Dhall. Skin segmentation dataset. *UCI Machine Learning Repository*, 2010.

[6] C. Craye and F. Karray. Driver distraction detection and recognition using RGB-D sensor. *arXiv preprint arXiv:1502.00250*, 2015.

[7] N. Das, E. Ohn-bar, and M. M. Trivedi. On Performance Evaluation of Driver Hand Detection Algorithms : Challenges , Dataset , and Metrics. 2015.

[8] A. Eriksson and N. A. Stanton. Takeover time in highly automated vehicles: Noncritical transitions to and from manual control. *Human factors*, 59(4):689–705, 2017.

[9] S. S. Farfade, M. Saberian, and L.-j. Li. Multi-view Face Detection Using Deep Convolutional Neural Networks. *Cornell University Library*, page 37, 2015. ISSN 0920-5691. doi:10.1007/s11263-015-0816-y. URL `http://arxiv.org/abs/1409.0575{%}5Cnhttp://arxiv.org/abs/1502.02766`.

[10] A. Fernández, R. Usamentiaga, J. L. Carús, and R. Casado. Driver Distraction Using Visual-Based Sensors and Algorithms. 16(1805):1–44, 2016. doi:10.3390/s16111805.

[11] T. Hoang Ngan Le, Y. Zheng, C. Zhu, K. Luu, and M. Savvides. Multiple Scale Faster-RCNN Approach to Driver's Cell-Phone Usage and Hands on Steering Wheel Detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 46–53, 2016. ISSN 21607516. doi:10.1109/CVPRW.2016.13.

[12] M. Koestinger, P. Wohlhart, P. M. Roth, and H. Bischof. Annotated Facial Landmarks in the Wild: A Large-scale, Real-world Database for Facial Landmark Localization. In *First IEEE International Workshop on Benchmarking Facial Image Analysis Technologies*, 2011.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[14] G. Li, Haoxiang and Lin, Zhe and Shen, Xiaohui and Brandt, Jonathan and Hua. A Convolutional Neural Network Approach for Face Identification. *Cvpr*, pages 5325–5334, 2015. ISSN 1063-6919. doi:10.1109/CVPR.2015.7299170. URL `http://users.eecs.northwestern.edu/{~}xsh835/assets/cvpr2015{_}cascnn.pdf`.

[15] S. Martin, E. Ohn-bar, A. Tawari, and M. M. Trivedi. Understanding Head and Hand Activities and Coordination in Naturalistic Driving Videos. 2014.

[16] E. Ohn-bar and S. Martin. Driver hand activity analysis in naturalistic driving studies : challenges , algorithms , and experimental studies challenges , algorithms , and experimental studies. *Journal of Electronic Imaging*, 22(4), 2013. doi:10.1117/1.JEI.22.4.041119.

[17] E. Ohn-bar and M. Trivedi. In-Vehicle Hand Activity Recognition Using Integration of Regions. *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pages 1034—-1039, 2013.

[18] E. Ohn-bar, S. Martin, A. Tawari, and M. Trivedi. Head , Eye , and Hand Patterns for Driver Activity Recognition. *22nd International Conference on Pattern Recognition (ICPR), 2014*, pages 660—-665, 2014. doi:10.1109/ICPR.2014.124.

[19] W. H. Organization. *Global status report on alcohol and health 2014*. World Health Organization, 2014.

[20] S. L. Phung, A. Bouzerdoum, and D. Chai. Skin segmentation using color and edge information. *Proceedings - 7th International Symposium on Signal Processing and Its Applications, ISSPA 2003*, 1(July):525–528, 2003. doi:10.1109/ISSPA.2003.1224755.

[21] T. M. Pickrell, H. R. Li, and S. KC. TRAFFIC SAFETY FACTS, 2016. URL https://www.nhtsa.gov/risky-driving/distracted-driving.

[22] L. Rokach. Ensemble-based classifiers. *Artificial Intelligence Review*, 33(1):1–39, 2010.

[23] U. D. o. H. &. H. Services. Distracted Driving, 2016. URL https://www.cdc.gov/motorvehiclesafety/distracted{_}driving/.

[24] K. Seshadri, F. Juefei-xu, D. K. Pal, M. Savvides, and C. P. Thor. Driver Cell Phone Usage Detection on Strategic Highway Research Program (SHRP2) Face View Videos. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 35—-43, 2015.

[25] I. Sultan. Academic purposes?, 2016. URL https://www.kaggle.com/c/state-farm-distracted-driver-detection/discussion/20043{#}114916.

[26] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[27] Tensorflow. TensorFlow-Slim image classification library. URL https://github.com/tensorflow/models/tree/master/slim.

[28] C. Yan, F. Coenen, and B. Zhang. Driving Posture Recognition by Joint Application of Motion History Image and Pyramid Histogram of Oriented Gradients. *International Journal of Vehicular Technology*, 2014, 2014.

[29] C. Yan, F. Coenen, and B. Zhang. Driving posture recognition by convolutional neural networks. *IET Computer Vision*, 10(2):103–114, 2016. ISSN 1751-9632. doi:10.1049/iet-cvi.2015.0175.

[30] S. Yan, Y. Teng, J. S. Smith, and B. Zhang. Driver behavior recognition based on deep convolutional neural networks. *12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pages 636—-641, 2016. doi:10.1109/FSKD.2016.7603248. URL http://ieeexplore.ieee.org/document/7603248/.

[31] X. Zhang, N. Zheng, F. Wang, and Y. He. Visual Recognition of Driver Hand-held Cell Phone Use Based on Hidden CRF. *2011 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, pages 248–251, 2011.

[32] C. Zhao, B. Zhang, J. Lian, J. He, T. Lin, and X. Zhang. Classification of driving postures by support vector machines. *Proceedings - 6th International Conference on Image and Graphics, ICIG 2011*, (June 2014):926–930, 2011. doi:10.1109/ICIG.2011.184.

[33] C. Zhao, Y. Gao, J. He, and J. Lian. Recognition of driving postures by multiwavelet transform and multilayer perceptron classifier. *Engineering Applications of Artificial Intelligence*, 25(8):1677–1686, 2012.

[34] C. H. Zhao, B. L. Zhang, J. He, and J. Lian. Recognition of driving postures by contourlet transform and random forests. *IET Intelligent Transport Systems*, 6(2):161–168, 2011.

[35] C. H. Zhao, B. L. Zhang, X. Z. Zhang, S. Q. Zhao, and H. X. Li. Recognition of driving postures by combined features and random subspace ensemble of multilayer perceptron classifiers. *Neural Computing and Applications*, 22(1):175–184, 2013.