PROJECT PROPOSAL: DRIVER DROWSINESS DETECTION

Daniel Liu
Student #1007028841
daniell.liu@mail.utoronto.ca

Srinidhi Shankar
Student # 1006994460
srinidhi.shankar@mail.utoronto.ca

Dante Crescenzi
Student # 1006938432
dante.crescenzi@mail.utoronto.ca

Leon Zhu
Student # 1006877603
leon.zhu@mail.utoronto.ca

ABSTRACT

Driver drowsiness leads to thousands of injuries and hundreds of fatalities annually, nationwide. This paper proposes a deep learning based approach to classify whether drivers are drowsy from videos of their faces. In related work, convolution neural networks have shown success in the extraction and classification of facial features from images. In this project, we propose a hybrid CNN-RNN architecture for the task of extracting and classifying facial features from videos instead. We then propose to alert drivers when they are drowsy to prevent further accidents. We discuss potential ethical considerations for our project as well as a long-term timeline for the execution of it.
—-Total Pages: 7

1 INTRODUCTION

Our project for APS360 this term will focus on an AI that aims to detect driver drowsiness. Our goal is to stop drivers from operating vehicles where there are clear signs of drowsiness, specifically yawning, as they pose a safety risk to themselves and everyone around them. The National Highway Traffic Safety Administration estimates that every year, there are about 100,000 drowsy-driving crashes that cause 800 fatalities and 50,000 injuries (Rivelli & Kempken). Eventually, our software aims to minimize this, by scanning the driver's face and sending notifications to advise them to stop driving if they start yawning, as yawning is one of the biggest signs of drowsiness (dro). To develop this software, we hope to leverage the high accuracy of facial classification through deep learning to train a neural network model that detects whether a driver is displaying signs of tiredness (Teoh et al., 2021). In this document, we will talk about the related work for our project, highlight our proposal, develop our project plan, and explain our risk register and ethical considerations.

2 BACKGROUND & RELATED WORK

Contemporary facial recognition techniques have shown significant improvement at extracting facial features through CNN methods in recent years. In this section, we briefly discuss the improvements on CNNs extracting facial features as well as their implementations.

CNNs trained by standard back-propagation have been shown to achieve spectacular accuracy when training on large datasets (Krizhevsky et al., 2012). Scientists at the Meta AI group then closed the gap in facial recognition to almost human-level accuracy with DeepFace which represented a face model in three dimensions (Taigman et al., 2014). DeepFace is a nine-layer deep CNN that trained on four million facial expressions of 4000 subjects. The first three layers are combined to make a single convolution-pooling-convolution filter to extract low level features like simple edges and texture. The next three layers are locally connected which also apply a filter bank with every location in the feature map learning a different set of filters. The final top two layers are fully connected with each output unit connected to each input unit. Taigman et al. (2014) advanced the extraction of facial features from CNNs only receiving 2D images as input to be able to receive out-of-plane images using a 3D alignment method.

Further applications of CNN models were then seen for predicaments such as the COVID-19 pandemic (Mundial et al., 2020). This work used a CNN model on the VGGFACE2 dataset with a total of 9500 different classes and 330 images per class. The model trained on this dataset to generate specific features of a human face. The model then used a masked dataset containing 800 images of 200 individuals. 200 images were unmasked and 600 images were masked. The classifier was then ultimately built with a support vector machine in the end. The trained model on the initial VGGFACE2 dataset had accuracies of 98%. This work also showcased that training on the masked dataset was necessary. The proposed model without training on the masked dataset only had an accuracy of 79%, however with training on the masked dataset, the accuracy increased to 97%. This shows that applications of CNN models are able to not classify only static facial features but also when a change such as a mask exists as well.

In this work, we use existing evidence of CNNs and their promise on the extraction and classification of facial features in images to motivate our work on alerting drivers when they are potentially drowsy. We explore the potential of a hybrid CNN-RNN neural network and make systematic evaluations of our model with a baseline.

3 Our proposal

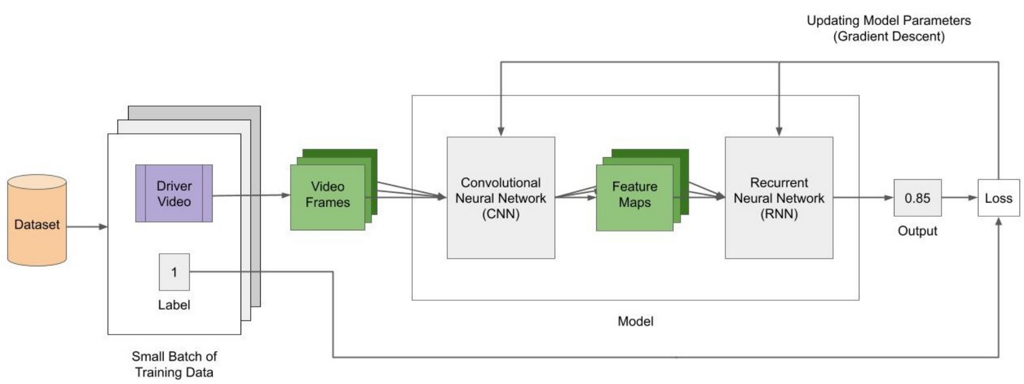

Figure 1: Illustration of Our Model/Training Workflow

3.1 Architecture

We wish to explore the potential of a hybrid CNN-RNN architecture for our video facial classification task. The intuition behind this architecture is to use CNNs to extract facial features from individual video frames and then to use RNNs to analyse the temporal progression of facial features across the entire video, detecting a yawn.

The input, in the format of a video, will be separated into its frames. Each frame will be passed into a series of 2D Convolutional Neural Network layers with the aim of generating a lower-dimensional representation of the frame's key features (feature map). Each frame's feature map will then be processed following the original sequence of frames in the video, using a Recurrent Neural Network, to consolidate each frame back into the original video. The final Recurrent Neural Network output after processing every frame's feature map will be the final classification output of the entire model.

3.2 Baseline Model

The features on the human face can be located by means of ratios. The mouth of somebody, for instance, is located roughly in the center and 38% of the way up the face measuring from chin to top of forehead (Bottino & Laurentini, 2010). Yawning, a major sign of drowsiness, often causes a person to open their mouth, which often results in the exposure of the darker inner mouth. Thus, a crude baseline solution can be crafted from sampling pixel values from the approximate location of the mouth of the person and checking if the samples are darker than a certain threshold. Figure 2 visually shows what will be done.

Cropping a frame of a video to center the face vertically and place the horizontal borders from the bottom of the chin to the top of the forehead will allow a program to then calculate the approximate location of the mouth of the person by means of ratios (Bottino & Laurentini, 2010). The pixels around the general area of the mouth can then be sampled to get an average darkness around the mouth area. This value can then be compared to samples taken from everywhere but the mouth area, to determine if the mouth area is darker. If it is by a certain margin, then we can infer that the person is yawning and therefore drowsy.

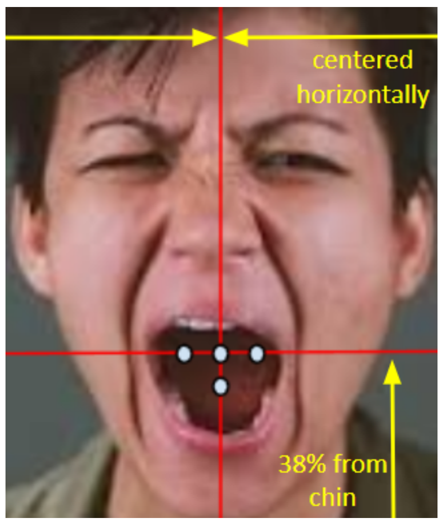

Figure 2: The uniform proportions of the face allow for the location and sampling of the mouth area.

3.3 Data Processing

The data will come from the public dataset 'Yawning Detection Dataset' that can be found on IEEE Dataport (Abtahi et al., 2020). The data set consists of videos of subjects yawning while driving their cars. The neural net would take in chunks of the video at a time in the form of frames of the video in order, so a script could be made to group the videos into batches of a certain number of frames. The script could, for example, split up the video into frames and rename the images to img_BB_XX where BB is the batch number and XX is the images place in the batch.

Another step of processing will be needed to prepare the videos for the baseline solution. The videos will need to be separated into their frames for pixel sampling. Then, the frames will need to be standardized to contain only the faces of the subjects, would have to be centered and cropped from forehead to chin. this can be easily achieved by means of Python's 'face-utils' library. Face-utils provides functionality to automatically crop an image of a person to the aforementioned state, perfect for the standardization that the baseline solution needs. An example and accompanying code is shown in figure 3 (The Python Package Index, 2021). A script can be made to splice and standardize all of the videos in the dataset and store their frames in a separate location with a prefix such as 'CROPPED' before each image name. This standardized dataset can then be used to validate the baseline solution.


```
import cv2
from face_utils.detection import Detector
from face_utils.cropping import cropping
imgPath = "files/Obama.jpg"
img = cv2.imread(imgPath)
img= np.array(img)
model = Detector("hog")
bbox = model.detect(img,2)
face = cropping.crop(img,bbox,1,80,80)
```

Figure 3: The automatic cropping of a photo of a person to only their face, with accompanying code

4 Project Plan

Our team plans on working together using Git as our version control system. The code and data will be stored on a private Github repository. Communication between group members on their specific code file will be practiced throughout the semester to avoid merge conflicts as well as overwriting each other's code. The team also plans to meet once a week to go over what they have worked on that week as well as their next steps. See Table 1 for more details on specific deadlines.

Table 1: Project Plan

Deliverable	**Tasks**	**Internal Deadline**	**External Deadline**	**Member**
Project Proposal	Introduction and Ethical Considerations Background and Project Plan Architecture and Illustration Data Processing and Baseline Model Risk Register	October 11, 2022	October 14, 2022	Srini Daniel Leon Dante Leon and Dante
Baseline Model Creation	Creation of the baseline model	October 17, 2022	N/A	Dante
Architecture Creation	Rough Creation of the primary neural network architecture	October 18, 2022	N/A	Leon Daniel
Training the model	Parsing of Data-set Fine-tuning hyperparameters Parameter changes Number of hidden layers Different loss criterion Different Activation Functions Validation Error Checking Testing Error Checking	October 31, 2022	N/A	All
Progress Report	Brief Project Description Individual Contributions/Responsibilities Data Processing Baseline Model Primary Model	November 2nd 2022	November 4th 2022	Srini All TBD Dante Leon or Daniel
Refine model	Parsing of Data-set Fine-tuning hyperparameters Parameter changes Number of hidden layers Different loss criterion Different Activation Functions Validation Error Checking Testing Error Checking	November 18 2022	N/A	All
Presentation Video	Problem Data Data Processing Model Demonstration Quantitative Results Qualitative Results (Optional) Takeaways	November 23 2022	November 25 2022	Srini Dante Dante/Leon Leon Srini Daniel Daniel All
Final Report	Contents of Final Report Other tasks TBD	November 30 2022	December 2 2022	All

5 Risk Register

A risk to consider is group member burnout. Engineering is a demanding program and many people need breaks at times. With this project commanding not just constant report deadlines but the actual development of the proposed NN all paired with other courses, there is a chance that a group member may become extremely busy and/or burnt out. In this case, the group has decided that if a group member needs a small break less work will be delegated to them which they will make up for in other less intensive ways, such as proofreading reports or updating the team wiki. When the member feels ready, they can get back to full speed with the team. The likelihood of this risk is moderate, as while all group members are to follow a well planned out internal deadline structure with built in time buffers, unforeseen events can always complicate situations.

Another risk to consider is if a team member does not complete their work before an agreed upon internal deadline. The likelihood of this risk is quite high, due to unforeseen circumstances or due to overwhelming amounts of other schoolwork and life commitments. To mitigate the effects of this risk, our team has decided to set internal deadlines in such a way that we complete each course deliverable at least 72 hours before its due date. Each team member has also agreed to try their best to let the team know 24 hours before an internal deadline if they feel it cannot be met. These rules will give enough leeway to either give the team member more time to complete their work, or to reorganise and redistribute the uncompleted work to other team members. For repeat offenders who fail to meet internal deadlines 3 times or more, we feel that this is an issue that is appropriate to bring up to the teaching team.

A final risk that can occur is team members disagreeing on a major decision, preventing the project from progressing. The likelihood of this risk is quite high. Each team member is very motivated to do well on this project and has strong opinions on how the project should be done. To resolve such a situation, the team must be active in ensuring that all members feel equally represented in each of the team's decisions. The team feels that the correct way to respond to such a situation is to open, honest, and constructive communication within the team. When a disagreement arises, the team will follow a structured discussion format where each team member will be invited to talk uninterrupted about their opinion on the subject, avoiding personal attacks and remaining constructive. Afterwards, the team will go into open discussion during which the points of each member will be argued. If a unanimous decision is not reached after 30 minutes of discussion, the team will go into a majority vote. In the event of a tie, the team will seek a third opinion from the teaching team.

6 Ethical Considerations

As our facial recognition model would rely on training data with videos of various people, there are important considerations with respect to diversity in our pool of data. Data from many different racial groups, ethnicities and genders would need to be collected to ensure that the software would work on anyone that would need to use it. This is a large ethical consideration with biometric software worldwide, as algorithms developed by IBM and Microsoft showed that these biometrics performed the worst on darker skinned females, with error rates of up to 34% higher than light skinned males. See Figure 4. (Najibi). Additionally, as the idea behind the software would be to embed it into vehicles, there are privacy considerations for devising a software that would be constantly recording a driver. Users could feel as if they are subjected to unwanted data collection, and to combat this, we could either make our software open source, or make sure to run our code locally instead of sending it to a cloud. It is also important to let users know what biometric data is being collected and get their consent before the use of the device.

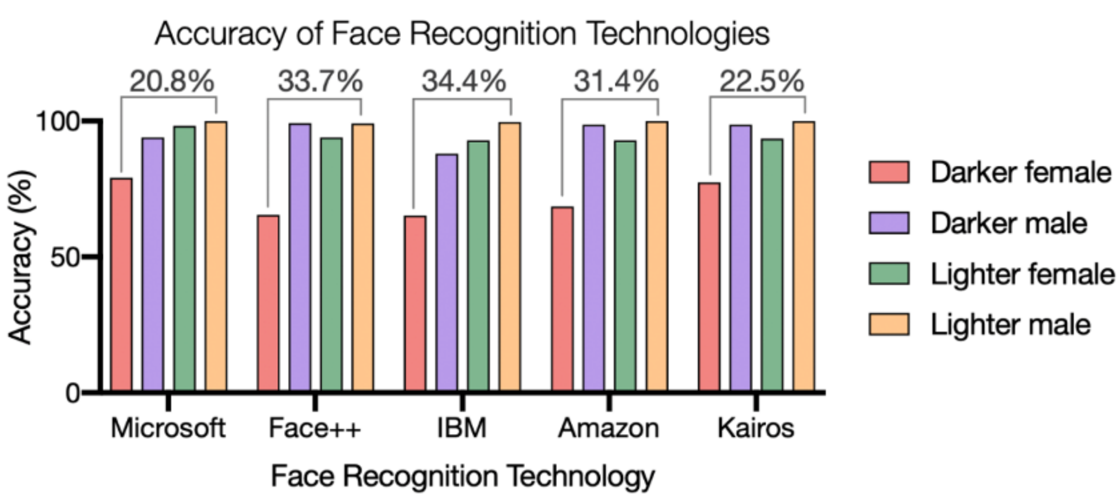

Figure 4. Graph showing the accuracy of facial recognition softwares, as tested by Microsoft, Face++, IBM, Amazon, and Kairos.

7 Source code

`https://gitfront.io/r/user-6785826/fik3Wsb3xgDo/aps360-proj/`

REFERENCES

Drowsy driving: Asleep at the wheel. *Centre for Disease Control and Prevention*.

Shabnam Abtahi, Mona Omidyeganeh, Shervin Shirmohammadi, and Behnoosh Hariri. Yawdd: Yawning detection dataset, 2020. URL `https://dx.doi.org/10.21227/e1qm-hb90`.

Andrea Bottino and Aldo Laurentini. The analysis of facial beauty: An emerging area of research in pattern analysis. pp. 425–435, 06 2010. ISBN 978-3-642-13771-6. doi: 10.1007/978-3-642-13772-3_43.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL `https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf`.

Imran Qayyum Mundial, M. Sohaib Ul Hassan, M Islam Tiwana, Waqar Shahid Qureshi, and Eisa Alanazi. Towards facial recognition problem in covid-19 pandemic. In *2020 4rd International Conference on Electrical, Telecommunication and Computer Engineering (ELTICOM)*, pp. 210–214, 2020. doi: 10.1109/ELTICOM50775.2020.9230504.

Alex Najibi. Racial discrimination in face recognition technology. *Harvard University, The Graduate School of Arts and Sciences*.

Elizabeth Rivelli and Maggie Kempken. Drowsy driving statistics and facts. *Bankrate*.

Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1701–1708, 2014. doi: 10.1109/CVPR.2014.220.

KH Teoh, RC Ismail, SZM Naziri, R Hussin, MNM Isa, and MSSM Basir. Face recognition and identification using deep learning approach. *Journal of Physics: Conference Series*, 1755(1): 012006, feb 2021. doi: 10.1088/1742-6596/1755/1/012006. URL `https://doi.org/10.1088/1742-6596/1755/1/012006`.

The Python Package Index. Example of face-utils, 2021. URL `https://pypi.org/project/face-utils/`. [Online; accessed October 11, 2022].