Drunk Detection using Machine Learning Methods and Infrared Images

Kevin Tang
kvntng@umich.edu

Andrew Vernier
amverni@umich.edu

1 Introduction

Alcohol consumption is known to have physical affects on the body which include changing a person's body temperature (Meyers, 2002). Studies have shown that this includes how different parts of the face's temperature vary relative to each other. This relationship is distinct enough that researchers can predict drunkenness using infrared images of human faces and hands. However, to date, drunk identification approaches have relied at least partially on a relatively slow and expensive human evaluation process. Different regions of the face need to be manually selected for analysis (Koukiou and Anastassopoulos, 2017). These individual regions are then extracted and used as features for several region-specific machine learning methods to provide an aggregate classification result. Thus, our goal is to create a system to automate the process of evaluating drunkenness from thermal images by utilizing machine learning methods to evaluate an entire image of the face and providing a classification result. This will allow for real time predictions and eliminate the need for experts to evaluate and define facial features in the thermal images.

Creating such a system would allow drunkenness prediction systems to be installed in cars in to prevent drunk driving. In 2018, 10,511 people were killed as a result of drunk driving crashes (NHTSA, 2019) — almost a third of traffic-related deaths in the United States stem from alcohol-related incidents (CDC, 2019). Drunk driving is a problem that we need to face. Even at the minimum blood alcohol content (BAC) where drivers are legally considered intoxicated, drunk drivers are twice as likely to be involved in traffic accidents than sober drivers (Zhao et al., 2014). People who are intoxicated should not be handling any motor vehicles or heavy machinery.

Methods do currently exist for detecting drunkenness that are currently used by law enforcement such as breathalyzers, blood tests, or urine tests (BACtrack, 2015). However, these methods can only be applied retroactively when an accident or reckless driving has already occurred. It is true that the aforementioned methods (e.g. requiring a breathalyzer test in order to start a car) can aid in enforcing laws against drunk driving, but they are not sufficient because they are considered too invasive to be required for all people (i.e. they are only applied to cars after a driver gets a DUI). Additionally, they cannot continuously detect drunkenness in their current implementations.

Previous research has looked into more robust drunkenness detection systems. One area of research includes using thermal images and experts to make predictions (Koukiou and Anastassopoulos, 2017) (Hermosilla et al., 2018) however this also cannot be deployed to make predictions continuously. Another area is attempting to use machine learning to make predictions based on RGB images (Takahashi et al., 2015) or speech (Bone et al., 2014) however these rely on features that are less reliable human indicators than thermal data (i.e. people exibit more variation in RGB images and speech).

There does not currently exist a method of predicting drunkenness that focuses on reliable human indicators (e.g. infrared images, blood tests, etc.) and employs automated methods. In order to do so, technology that can subtly detect drunkenness before and during driving must be utilized so as to ensure a safe driving environment for everyone in and around cars.

Our proposed solution is to use an infrared camera to capture images of the driver's face to evaluate sobriety through machine learning algorithms and grant the ability to start the car's engine. Additionally, since the images can be periodically taken while driving, this solution can also continuously evaluate sobriety in order to prevent a driver from

driving if they become intoxicated at a point after starting the car. It is out of the scope of this project to say what a car would do in such a situation, but a critical step in limiting drunk driving as much as possible is detecting drunkenness. Our project implements various machine learning models which will classify the infrared images as coming from a drunk or sober individual. Thus, while not solving the entire problem, our solution is a necessary piece of solving our ultimate goal of increasing human safety on and around roads.

The system required to implement this would include an infrared camera directed towards the driver's face and an onboard computer to process and classify infrared images. The system's computer would have a pre-trained model which would take infrared images as input and output sobriety predictions. It would evaluate the sobriety of a driver when he/she turns on the vehicle and would re-evaluate this periodically throughout a drive since it is known that a person is not immediately impacted by alcohol consumption and is also able to consume alcohol while driving. The major drawback of this solution is the relative accuracy when compared with the methods that are currently employed for determining sobriety after a crash, like breathalyzers or blood tests. However, it is largely unknown exactly how accurate such a system could be, so this project is partially aimed at determining the potential feasibility of this solution in a real world setting.

Our experiments support the claim that we can use machine learning algorithms to analyze infrared facial images of humans to predict drunkenness. Our best classifier was a single layer CNN which obtained an accuracy of 0.87. We note that our experiments were performed on a small dataset (164 images, 41 human subjects) that was augmented through various methods explained in Section 3. Thus, we note that the significance of our results is not to say that this method of predicting drunkenness should necessarily be used in the future. Rather, we think these results show that using a CNN to predict sobriety based on infrared images is a promising idea that should be researched more thoroughly using a larger dataset that is representative of the general population. These results support the claim that it is possible to obtain a continuous and real-time system for automating the process of evaluating the sobriety of drivers in order to prevent drunk driving which is something that does not yet exist.

The remainder of this paper presents more details on our research and findings. First, we look at related work and how our system differs from such works. Then, we discuss our dataset followed by how we analyze the data. Next, we discuss the algorithms we implemented to process the data. This is followed by a discussion of our key results. Then we delve into the ethical considerations of our proposed solution. Lastly, we highlight the major conclusions that result from our research.

2 Related Work

As we discussed in the Introduction section, all of the methods of evaluating sobriety that are currently used in practice have a few key disadvantages. These methods include breathalyzers, blood tests, and urine tests and are considered to be the most reliable methods. First, because these require the supervision of the authorities, they are viewed by much of society as being too invasive for us to enforce that every individual uses them to prove sobriety every time someone drives. Thus, these systems are only used to check if a person was driving under the influence after police have reasonable suspicion or to allow a person with previous DUI offenses to turn on their car engine. Furthermore, even if we did force everyone to use these systems to be allowed to drive, it does not solve the problem of checking for drunk driving after the car is started.

Thus, there has been much research effort put into finding a more robust system for preventing drunk driving. The first area of research that has been of great interest is determining what human factors can be used to reliably predict drunkenness. Studies have shown that the human face exhibits different thermal characteristics when a person has consumed alcohol. These characteristics are distinct enough that thermal infrared cameras can be used to evaluate a person's face—specifically the forehead, nose, and cheeks—in order to discriminate between a sober and drunk person. In fact, a study at the University of Patras has shown that an intoxicated person can be identified using only a thermal infrared image of his or her face while intoxicated; there is no need for images of the same person's face while sober. This study was able to predict drunkenness with 0.85 accuracy using just the thermal images of an individual's face. However, this study used humans who were

trained in extracting facial features in a consistent way from infrared images to annotate bounding boxes on regions of interest (e.g. the forehead, cheek, etc.) before applying local difference patterns (Koukiou and Anastassopoulos, 2017). This is not a reasonable solution to the problem at hand since these manual steps do not allow the solution to be scaled. However, this research does bring to light the promise of using infrared images as a reliable indicator for drunkenness.

Another prominent area of research is using machine learning to automate the process of predicting drunkenness. One common method is using RGB images. Studies have shown that this can be quite reliable for predicting drunkenness (Takahashi et al., 2015). A second method is using speech as the primary indicator. Studies have shown that this can achieve an accuracy of up to 0.72 (Bone et al., 2014). Both of these methods fail when they attempt to predict on certain people who have certain facial features or speech patterns that the machine learning algorithm learned to be an indicator of drunkenness. Additionally, both of these methods can continuously evaluate drunkenness throughout driving although a driver is not forced to speak while driving. These methods are popular because they match the intuition that one can "spot" a drunk person by how a person looks or how a person's voice sounds. However, these ideas rely on us having prior knowledge of what a person normally looks or sounds like which are things that a machine learning system like this cannot rely on. Additionally people have different tolerances to alcohol and are affected by alcohol differently, so these methods quickly become unreliable.

Our solution attempts to combine the best of both of these two methods: we take the infrared images that were deemed to be a good indicator and create a prediction system that is automated end-to-end using machine learning approaches. This utilizes the reliability of people's thermal reactions to alcohol consumption and the ability for machine learning approaches to be automated and make predictions continuously. Thus, we created a machine learning model to predict drunkenness using infrared images.

3 Data

We will be using the frontal facial infrared images in the SOBER-DRUNK DATA BASE from the University of Patras (Koukiou and Anastassopoulos, 2017). The SOBER-DRUNK DATA BASE consists of infrared images and sobriety labels from 41 different people; each subject has 1 frontal facial image while sober and 3 at various times after consuming alcohol. This amounts to 164 unique thermal infrared images. We have split this data into train (approximately 65% of the data), test (approximately 20% of the data), and validation (approximately 15% of the data) sets. Since there were multiple images per subject, special care was taken to ensure that a none of a subject's images would not bleed into other sets. All images from the same subject appear in only one of the splits to emulate how this system would perform in the real world; nearly all individuals that this system would evaluate when deployed will not be subjects that the model will be trained on. Since approximately 75% of our data has the label drunk, we are using 0.75, or chance, as the baseline accuracy to compare our models' performances to.

The images in the SOBER-DRUNK DATA BASE do not specify the BAC, but rather have binary sober or drunk labels. Therefore, we do not have sufficient data to train a model to predict BAC based on thermal infrared images of the face. Further, our definition of drunk is having consumed 4 servings of alcohol within the past 1.5 to 2.5 hours. This matches the methodology used to collect the data in the SOBER-DRUNK DATA BASE. The exact methodology used for data collection is specified in Appendix A Figure 7.

Since the number of images to train from was relatively small, the train set was augmented using different types of noise and transformations to create additional training data. For every image in the original set, we created three new images so that our training set had a total of 104 images. The first image augmentation was a mirror of the original with a very small amount of added Gaussian noise ($\mu = 0$, $\sigma^2 = 9$). The second image augmentation was the original image with a small amount of Gaussian noise ($\mu = 0$, $\sigma^2 = 25$). The third image augmentation was the original image that had an average blur applied. We added a blur by setting each pixel value to be the average of value of the 3×3 grid centered at the corresponding pixel in the original image. All of these values and distribution types were empirically determined while keeping in mind that we did not want to create images that differed too drastically from the originals. Table 1 compares the original image with its three

corresponding augmented images.

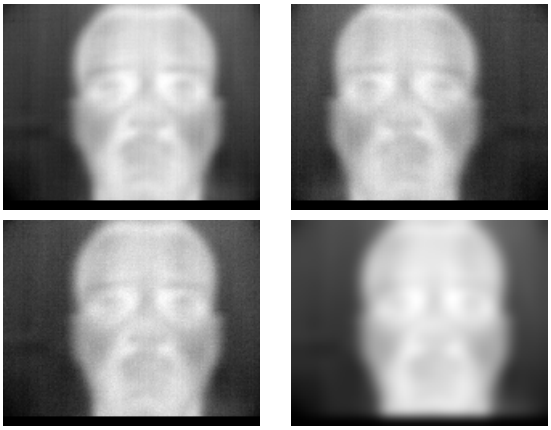

Table 1: Original image (top-left), flipped image (top-right), noisy image (bottom-left), and blurred image (bottom-right)

4 System Architecture

Our system will consist of three components. One for data preprocessing, one for model training, and one for making predictions. These components are intended to be used sequentially but are broken apart for convenience. For example, we are now able to run our data processing component once and then use this data for multiple different implementations of the model training component (i.e. various machine learning algorithms).

4.1 Data Preprocessing Component

The first component is the data preprocessing component which will take in the raw thermal infrared image data from our dataset (50 frame, 128x160 pixel TIFF files). We will only use the infrared images of human faces (the remaining infrared images will be discarded in this step). This component will then perform tasks such as data augmentation and normalization. The data augmentation is discussed in more detail in Section 3. The output will be preprocessed image data in a pickle file. Figure 1 shows a high-level diagram of this component.

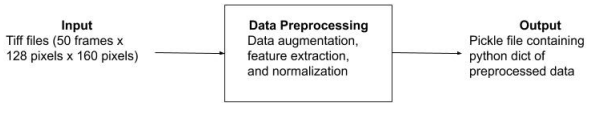

Figure 1: Data Preprocessing Component Diagram

4.2 Model Training Component

The second component will perform model training. The input will be the output of the first component – a pickle file containing preprocessed image data. The output will be a trained model stored in a pickle file. Figure 2 shows a high-level diagram of this component.

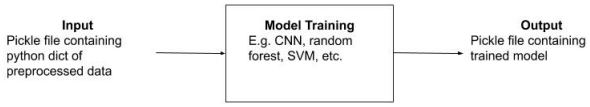

Figure 2: Model Training Component Diagram

The final model that we use is discussed in more detail in Section 6.

4.3 Prediction Component

The third component will make sobriety predictions. The input will be a frontal facial thermal image (i.e. a 50 frame, 128x160 pixel Tiff file) and a pickle file containing a pretrained model – the output of the second component. The output will be a binary prediction of either sober or drunk. This component will perform also preprocessing on the input image in order for the input format to match that of the model's training data. Figure 3 shows a high-level diagram of this component.

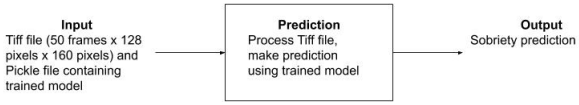

Figure 3: Prediction Component Diagram

5 Methods for Analyzing Data

After reading in the images from the SOBERDRUNK DATA BASE and augmenting our the dataset, we converted the images from 50-frame tiff images into a single layered representation. This turned the three-dimensional images into a flattened two-dimensional representation of the same data. We found that using the two-dimensional representation of the data led to a better performance of our final model. This process of converting the images is based off of the code described in the README of the dataset which is shown in Appendix A. Examples of the flattened representation are shown in Table 1.

One way we attempted to preprocess the original image to extract more useful information for classifying drunkenness was by detecting the face and extracting the rectangular region bounding the face. Once a face had been detected, the rectangular regions bounding facial features (e.g. the

forehead, cheeks, etc.) could be more easily detected and extracted to be used as features for our model. Extracting these facial features to use as features for machine learning methods may be useful for our application since there is evidence to believe the temperature of various facial features, specifically the nose and forehead, are good indicators of drunkenness (Koukiou and Anasassopoulos, 2013).

We attempted using multiple methods of standard face detection and facial feature extraction including a pre-trained Haar cascade classifier using OpenCV, the face-detection Python library, and a face detection API called Betaface API. We originally hypothesized that at least one of these methods could accurately and precisely bound the face and facial features. However, these methods of face detection and facial feature extraction were designed to operate on RGB images.Thus, we were not able to accurately and precisely find bounding boxes around the face and facial features for the infrared images contained in our dataset. This will be discussed further in face detection and Facial Feature Detection for Feature Extraction section in Appendix B.

6 Algorithms to Interact with Users through Drunkenness Predictions

We attempted to use various machine learning methods for making drunkenness predictions with an infrared facial image as input.These methods include simple classifiers, voting classifiers, and autoencoders which are discussed in detail in Appendix D.

The method which yielded the best performing classifier was using a CNN. Our base CNN, implemented in Keras, takes in a 2D input array (160×128) representing an infrared image of a human face and outputs a binary sobriety prediction. The final model had one convolutional layer and one dense layer. In order to find the structure for our final model, we choose several key model parameters for hyperparameter tuning: *number of additional conv2d layers, filter size of conv2d layers, number of additional dense layers, filter size of dense layers, learning rate, and batch size*. We then performed hyperparameter tuning with the aforementioned hyperparameters by implementing a grid search in order to produce various models with a range of classification performances. Each model was trained until validation accuracy did not increase within 30 epochs, where early stopping would occur. Following hyperparameter tuning, we used Tensorboard to compare accuracy and loss metrics between models produced to select the model with the best classification performance. Figure 4 shows an example hyperparameter tuning run with various accuracy and loss per model.

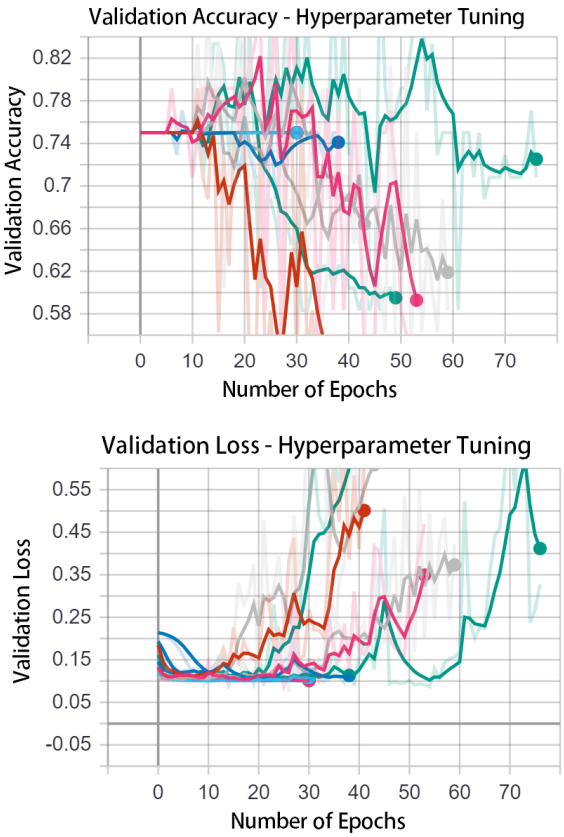

Figure 4: Tensorboard metrics were used to explore CNN model training behavior and assess which model from each hyperparameter run was best. Each individual model's validation accuracy and loss during training is denoted by it's own color.

7 Results and Discussion

Out of all methods tested, we found that a CNN worked the best to classify drunkenness with a test accuracy of 0.87 compared to the 0.75 baseline accuracy. The CNN with the best classification performance had one convolutional layer (8x8 filter) and one dense layer (512x512 filter), a learning rate of 0.01, and a batch size of 32. A model architecture visualization of this best performing CNN can be seen in Figure 5.

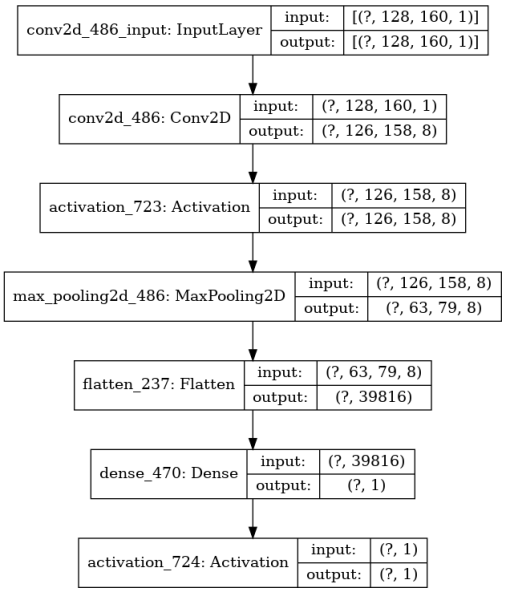

Figure 5: The model architecture of the best performing CNN found during hyperparameter tuning is displayed above. This CNN has one convolutional layer (8x8 filter) and one dense layer (512x512 filter) and was trained with a learning rate of 0.01 and a batch size of 32.

Looking more closely at the predictions, out of 70 test images, we obtained 7 true positives, 9 false positives, 0 false negatives, and 54 true negatives. A visualization of the previous test results can be seen in the confusion matrix shown in Figure 6.

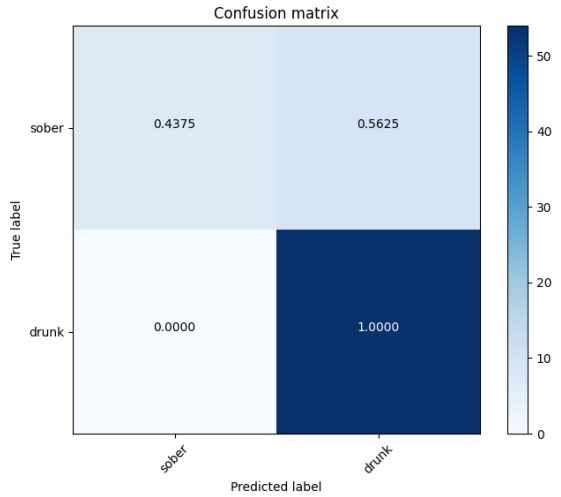

Figure 6: The confusion matrix visualizing the test results of the best found CNN. A total of 70 images were in the test set, leading to 70 predictions.

It is successful that we obtained 0 false negatives (i.e. drunk people who were classified as sober) since in this context, a false negative means our system would have let drunk drivers onto the road. However, obtaining 9 false positives (i.e. sober people who were classified as drunk) is not an ideal result since it can be disruptive and irritating to the users of the system if the user is wrongfully stopped from using their vehicle. We will discuss further the implications of misclassification in Section 8.2.

Next, we assessed if our final model's performance on the test set was statistically significantly better than a classifier that predicted every label to be drunk, our majority class. This second model corresponds to our baseline classifier discussed in Section 3 which is simply chance. To verify that our model's performance, which yielding 0.87 accuracy, was better than the baseline classifier, yielding our baseline 0.75 accuracy, we conducted a one-way ANOVA test using the two sets of predictions. Our null hypothesis was that our model's test set predictions performed equally well compared to the baseline classifier's test set predictions at a significance level of 0.05. The ANOVA test yielded $F-value = 3.76$ and $p-value = 0.0076$. Since $(p-value = 0.0076) < \alpha = 0.05$, we reject our null hypothesis. In conclusion, we have sufficient evidence to support that our CNN model predicts drunkenness better than the baseline classifier that predicted that all subjects were drunk.

8 Ethical Considerations

This section discusses the major ethical considerations that arise from attempting to use machine learning on thermal imaging as a means of preventing drunk driving. First we consider how this system may be an invasion on privacy. Then we discuss the implications of improperly predicting drunkenness. Lastly, we look at how this system may affect liability in the event of an accident.

8.1 Privacy Concerns

This system requires taking photos of a person multiple times while driving. Since this is collecting data from a user, we would need to get consent before doing so. Further, given our proposed use case, the system would only allow a person's car to turn on if the system classified the driver as being sober. Giving such a machine learning system this kind of power would be considered by some users to be intrusive. With this in mind, our proposed system would have to have enough societal gains from an increase in safety to overcome the added inconvenience and decrease in privacy that it brings.

8.2 Implications of Misclassification

Both false negatives, classifying a user as sober when they are drunk, and false positives, classifying a user as drunk when they are sober, can have serious negative implications on people's lives.

False negatives can lead to dangerous situations for the obvious reason that the system allowed a drunk driver on the road which was the exact problem this system was intended to prevent. False positives are bad for a slightly more subtle reason. They can lead to people not being allowed to drive when they need to. If a driver needs to take someone to a hospital and the system falsely does not allow them to, the system can again be putting people in serious danger. While the situation we just described is severe, it is admittedly not common. However, even false positives 10% of the time can lead to people being commonly late for work, etc. which can cause the unpopularity of this system.

Thus, our proposed system would likely need to strike a balance between prioritizing recall and precision. Our system needs to classify with a high degree of recall to prevent drunk drivers and dangerous situations. However, since there is a trade off between recall and precision, our system also needs to be precise enough as not to inconvenience users.

8.3 Liability

Under current societal norms, it is clear that if a person is caught driving with a BAC over the legal limit, that individual is responsible for their actions because they made the choice to drive on their own. However, if this system told a driver they had a BAC under the legal limit, and then they were pulled with a BAC above the legal limit, how much responsibility would fall on the individual as opposed to the system? Further, let's consider an individual who had a particularly low alcohol tolerance that decided they wanted to drive after having a drink. Let's also say that this individual would have normally decided not to drive because they knew how alcohol affected them but the system told them they had a BAC below the legal limit (and for this instance let's assume they really did have a BAC below the limit) so they figured it would be safe to drive. If this individual were to then get into an accident, would the system or the individual be responsible? It is not entirely clear.

9 Conclusions

Drunk driving is a serious issue that needs to be prevented to save countless lives from being unnecessarily taken. One reason drunk driving may be so prevalent is that in the status quo, any punishment for drunk driving is retroactive (i.e. punishment is only given when caught). This is due to the fact that the methods we use to detect drunkenness are either too invasive to force upon all drivers (e.g. breathalyzers needed to start car engines) or it is not feasible to scale up to all drivers (e.g. having an expert manually perform steps needed to detect drunkenness). Thus, it follows that subtle and automated drunkenness detection is the first step in the process of stopping drunk drivers from harming themselves and others.

Directed by research to suggest that facial temperature distributions are a reliable human indicator to predict drunkenness, we attempted to use various machine learning methods to classify infrared images as drunk or sober. Our results suggest that it is feasible to use a CNN classifier to predict if subjects in infrared images are drunk or sober with a relatively high accuracy (0.87). Moreover, we have found that it may not be necessary to extract specific regions of interest on the face (e.g. the forehead, cheeks, etc.) to maintain a high level of classification accuracy since it may be that a CNN training on an entire image of the face can establish relationships between regions of interest automatically. Our approach has the added benefit that the classification process can happen continuously and with relatively low latency since a pre-trained machine learning model can be used.

In our research, our model performance was limited by the size of the small dataset it was trained on. For future research, we predict that obtaining a larger dataset of infrared facial images would enable one to train a model to predict drunkenness with an even higher accuracy. Furthermore, our model performance was limited by the limited time and computing resources available for this project, since hyperparameter tuning for CNNs can take a long time without dedicated hardware. In future research, a larger dataset and more computing resources could yield better results. Lastly, we were not able to predict BAC since the data from the SOBER-DRUNK DATA BASE is labeled sober or drunk. We suggest that a future area of research that may be interesting is BAC prediction, since our results have demonstrated that CNNs can learn

the human indicators of drunkenness relatively accurately. A dataset of thermal infrared images of human faces, labeled with BAC at the time of photographing the subject, would be needed, however.

Our research into drunkenness detection for infrared images is ultimately directed by the possibility that infrared cameras may be used in the future to evaluate the eligibility of an individual to drive. When thinking about drunkenness detection in this context, we need to carefully consider the privacy concerns, implications of misclassifcation, and liability issues that arise with the use of this kind of technology. For example, users may not consent to being monitored, users that are misclassified may be wrongly prevented from driving – or worse, driving while intoxicated – and the liability of a resulting accident is unclear. Refer back to Section 8 for a more in-depth analysis of these ethical considerations. We have shown it is certainly possible to implement a drunkenness detection system that is automatically and continuously monitoring a driver. However, ultimately, more discussion needs to occur to collectively determine if such a system is ethical and whether the societal benefits exceed the costs incurred to the individual when deploying such a system on a wide scale.

References

BACtrack. 2015. Three types of bac testing.

Daniel Bone, Ming Li, Matthew P. Black, and Shrikanth S. Narayanan. 2014. Intoxicated speech detection: A fusion framework with speaker-normalized hierarchical functionals and gmm supervectors. *Computer Speech Language*, 28:375–391.

CDC. 2019. Impaired driving: Get the facts. *Centers for Disease Control and Prevention, National Center for Injury Prevention and Control*.

Gabriel Hermosilla, José Luis Verdugo, Gonzalo Farias, Esteban Vera, Francisco Pizarro, and Margarita Machuca. 2018. Face recognition and drunk classification using infrared face images. *Hindawi Journal of Sensors*, 2018.

G. Koukiou and V. Anasassopoulos. 2013. Face locations suitable drunk persons identification. In *2013 International Workshop on Biometrics and Forensics (IWBF)*, pages 1–4.

Georgia Koukiou and Vassilis Anastassopoulos. 2017. Local difference patterns for drunk person identification. *Multimedia Tools and Applications*, 77:1–13.

R.D. Meyers. 2002. Alcohol's effect on body temperature: Hypothermia, hyperthermia or poikilothermia? In *Brain Research Bulletin, Volume 7, Issue 2, August 1981*, pages 209–220.

NHTSA. 2019. 2018 fatal motor vehicle crashes: Overview. *Traffic Safety Facts Research Note*.

K. Takahashi, K. Hiramatasu, and M. Tetsuishi. 2015. Experiments on detection of drinking from face images using neural networks. In *2015 Second International Conference on Soft Computing and Machine Intelligence (ISCMI)*, pages 97–101.

X. Zhao, X. Zhang, and J. Rong. 2014. Study of the effects of alcohol on drivers and driving performance on straight road. *Modeling and Simulation in Transportation Engineering 2014*.

A SOBER-DRUNK DATA BASE README

The README file from the SOBER-DRUNK DATA BASE provides information regarding the contents of the dataset and how to flatten the infrared images from 50-frame tiff files (i.e. three-dimensional representations of the infrared image) into two dimensional representations of the image which can help with analyzing and processing the data. It also details the methodology for data collection. Figure 7 shows the README.

```
************************************************************

                SOBER  -  DRUNK   DATA BASE

(Started September 2012    ----   Completed April 2013)

Electronics Laboratory - Physics Department
University of Patras  -  Greece

By Georgia Koukiou and Vassilis Anastassopoulos
gkoukiou@upatras.gr      vassilis@upatras.gr

Every body can use this data base testing and publishing
experimental results, provided that she/he will refer to
relevant publications of the creators of this database.

************************************************************

It contains data for 41 persons.
For each person there are 16 different acquisitions
Each acquisition corresponds to each file of the data base
Each file contains 50 sequential frames of the same object
   acquired every 100msec, i.e. in 5 sec all 50 frames.


The following MATLAB program is provided for reading each
   separate file

************************************************************

clc;
clear all;
close all;

c=zeros(128,160);
for i=1:50
    a(i).data=imread('filename.tif',i);
    xm(i).data=min(min(a(i).data));
    a(i).data=(a(i).data-xm(i).data);
    for j=1:128
        for k=1:160
            c(j,k)=c(j,k)+a(i).data(j,k);
        end
    end
end

************************************************************
************************************************************

Infrared image acquisition

Time 20:50
Firstly, for each sober person, which is in calm condition,
an infrared sequence (1) is obtained from his Face (f), fro
his Eyes (e), from his Ear-profile (r), and his Hand (h).

21:00 - 22:00
After that four glasses of wine are drunk in one hour time.

22:20
A new sequence (2) of infrared images is acquired
    (f), (e), (r), (h).

22:50
A new sequence (3) of infrared images is acquired
    (f), (e), (r), (h).

23:20
A new sequence (4) of infrared images is acquired
    (f), (e), (r), (h).

We had the people in groups of 4 or 5 or 6 persons.
For two groups (10 people) we asked the police to carry
out measurement with alcohol-meter.
We have the correspondence of these measurements
with the persons.

************************************************************

Naming the files
serialnumber_personfirstname_acquisitionsequence_imageconte
_sex_age_weight__alcoholmeter

The two last measurements were obtained from few persons

************************************************************
```

Figure 7: SOBER-DRUNK DATA BASE README explains the contents, usage, and collection methodology.

B Facial Feature Detection for Feature Extraction

Extracting facial features to use as features for machine learning methods may be useful for our application since there is evidence to believe the temperature of various facial features, specifically the nose and forehead, are good indicators of drunkenness (Koukiou and Anasassopoulos, 2013).

We attempted to perform facial feature detection as a means of feature extraction. We tried multiple methods of standard face detection and facial feature extraction including a pre-trained Haar cascade classifier using OpenCV, the face-detection Python library, and a face detection API called Betaface API. However, after many attempts to use the previous methods, we were unsuccessful in extracting facial features in an accurate and precise enough manner to use these rectangular bounding boxes as features for machine learning methods in all methods. Occasionally, a correct bounding box would be drawn around the specified facial features. However, more times than not, the methods we tested either failed to detect any face or facial features in the infrared image or mislabeled regions of the face. Figure 8 below shows an example of a mislabeled resulted obtained using a Haar cascade classifier.

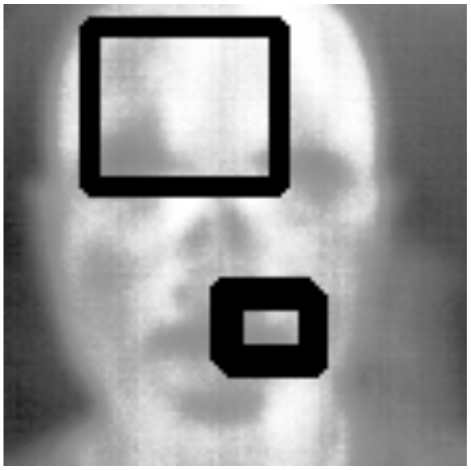

Figure 8: An example of mislabeled results produced by using a pre-trained Haar casacade classifier for the nose.

We originally hypothesized that at least one of these methods could accurately and precisely bound the face and facial features since standard face detection and facial feature extraction techniques operate on 1-layer 2D arrays (e.g. RGB image inputs are converted to grayscale), similar to the 1-layer 2D array formats of our infrared images.

However, after being unable to do so, we conjecture that it may be because these methods of face detection and facial feature extraction were trained on and designed to operate on RGB images, and thus do not generalize well to infrared images. There are key distinctions between the RGB/grayscale and infrared images (e.g. shadows exist in RGB/grayscale images but do not exist in infrared images) that make facial feature detection fundamentally different on the types of images. Thus, we were not able to accurately and precisely find bounding boxes around the face and facial features for the infrared images contained in our dataset.

C Generative Adversarial Networks for Data Augmentation

We attempted to use GANs as a means of augmenting our data. However, after many attempts to do so, we were unsuccessful in creating augmented images that were usable. Figure 9 shows an example of a single batch of the original images that were used to train the GAN and Figure 10 shows an example of the images generated by the GAN.

Figure 9: Example batch of real images used to train GAN for image augmentation.

It is clear that the generated images could not be used to augment our dataset because they do not resemble the original images in any way. If we represented the infrared images as RGB images, then we were able to get generative images that looked like they came from the original dataset. However, treating the infrared images like RGB images, then we lost 94% of the data because the RGB images only took into account the first three frames of the 50 frame TIFF files (these first three frames were treated like the red, green, and blue channels). Even though the generated images looked useful, they underlying data was not because when we would sum up the 50 frames of data (discussed in Section 5), the sum would be of only three frames and thus it would have much lower values than it should have. After researching why the GAN could not handle all 50 frames of the infrared image, our leading theory was that it was too much data for the GAN to use effectively given the resources we had (time, computational power, etc.) however we were unable to come to any firm conclusions on why this did not perform well.

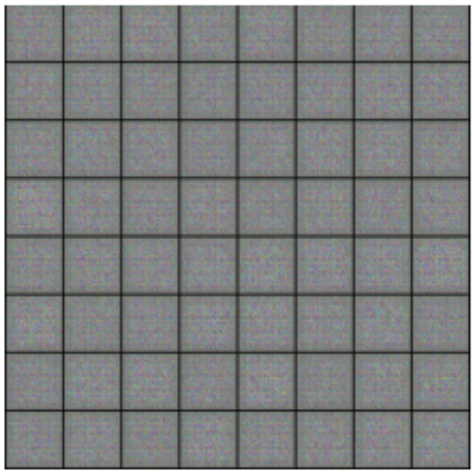

Figure 10: Example batch of images generated by GAN for image augmentation.

D Failed Classifier Attempts

This section discusses the various other types of classifiers that we attempted to use before settling on using a CNN. We began by trying a number of simple classifiers. Then, we tried to combine the power of these simple classifiers by using a voting classifier. Lastly, we tried using autoencoders.

D.1 Simple Classifiers

Our initial attempts at solving this problem were to use simple classifiers. The types of classifiers that we tried include SVMs, random forests, decision trees, logistic regression models, multi-layer perceptron models, K-nearest neighbor models, and stochastic gradient descent classifiers. We had initially planned on using the average temperatures of different facial regions as the features for these classifiers but we were unable to do that because we were unable to accurately and precisely perform facial feature extraction which is discussed

in more detail in Appendix B. Thus, the input features for these classifiers became flattened versions of the two-dimensional arrays that represented the summed thermal data from the images. In other words, we used a one-dimensional representation of the infrared image. This corresponded to the features for these classifiers being the summed thermal data for individual coordinate locations on the images. We theorize that our inability to extract more meaningful features from the images may have caused these classifiers to perform poorly. With each type of classifier, we performed hyperparameter tuning to find the best model for each type of classifier using a number of different model parameters. Table 2 summarizes the best obtained accuracy on the test set by each type of classifier and the corresponding p-value of a one-tailed T-test.

Classifier Type	**Best Accuracy**	**p-value**
Decision Tree	0.75	0.50
KNN	0.73	0.54
Logistic Regression	0.80	0.21
Perceptron	0.75	0.50
Random Forest	0.83	0.10
SGD	0.80	0.21
SVM	0.77	0.38

Table 2: Performances of simple classifiers

None of these classifiers performed well enough to support the claim that they performed better than chance at a 0.05 significance level.

D.2 Voting Classifier

Since each simple classifier performed poorly but still better than chance, we attempted to combine the power of each through a voting classifier. We took the best classifier of each type discussed in Appendix D.1 found from hyperparameter tuning and used them in a voting classifier. We also took subsets of these best simple classifiers to try to find the best voting classifier. The best voting classifier model that we were able to train had an accuracy of 0.84. This has a corresponding p-value of 0.08, so at a 0.05 significance level, we do not have sufficient evidence to support the claim that a voting classifier performs better than chance.

D.3 Autoencoders

We attempted to use autoencoders as a method of learning the data in order to perform classification. We tried various architectures for the autoencoders we tried by tuning parameters such as number of layers, number of filters, types of filters, and type of activation function. Ultimately we could not get an accuracy better than 0.77 which corresponds to a p-value of 0.38. At a significance level of 0.05, we cannot support the claim that our best autoencoder performs better than chance.