# Predicting Distracted Drivers

Tracy Cardwell

https://github.com/tcardwell/Predicting-Distracted-Drivers/blob/master/1_Data_Preprocessing.ipynb

https://github.com/tcardwell/Predicting-Distracted-Drivers/blob/master/2_Visualize_Data_Augmentation.ipynb

## Introduction

The National Highway Traffic Safety Administration (NHTSA) estimated that in 2018, driver distraction was a factor in about 8 percent of all fatal crashes and 15 percent of all injury crashes. Every day about 8 people were killed and more than 1000 injured in crashes that involved a distracted driver. Five percent of all drivers and eight percent of drivers 15 to 19 years old involved in fatal crashes were reported as distracted at the time of the crash. These crashes killed and injured not only drivers and passengers, but also pedestrians and cyclists.

These estimates may be low as it is challenging to determine distraction in driver fatality crashes, and self-reported distractions are likely underreported. Survey research indicates self-reporting of negative behavior is lower than actual occurrence of that behavior.

Distractions include any activity that takes your attention away from driving and fall in three main categories:

- visual: looking at something not related to driving;
- manual: taking your hands off the steering wheel; and
- cognitive: taking your mind off driving.

Texting combines all three types of distractions, making it especially dangerous. Sending or reading a text requires you to take your eyes off the road for about 5 seconds. If you are going 55mph, this means you are driving the length of a football field without looking.

A study conducted by Virginia Tech's Transportation Institute concluded that taking your eyes off the road for more than just two seconds doubles your risk of a crash.

A AAA Foundation report reviewing dozens of studies in 2008 concluded that any cell phone use, even hands-free, quadruples crash risk.

Many partial solutions have been adopted, including bans on texting and hand-held phone use, as well as do-not-disturb apps for phones. While these methods have slightly reduced accidents related to cell phone distractions, they are not enough. These measures cannot be completely effective as driver distractions predate cell phones. The problem is just more widespread now that cell phones are ubiquitous.

Systems that detect distractions and report them to drivers in real-time could further reduce distracted driving accidents and deaths. Insurance companies could offer such systems to drivers in exchange for a reduction in premium. They already have similar arrangements for systems that monitor driving habits including speed and cornering.

This project aims to identify driver distractions from dashboard camera images.

## Data Collection

The data analyzed in this project was downloaded from https://www.kaggle.com/c/state-farm-distracted-driver-detection. The data contains 10 classes of images as listed below.

- c0: safe driving
- c1: texting – right hand
- c2: talking on the phone – right hand
- c3: texting – left hand
- c4: talking on the phone – left hand
- c5: operating the radio
- c6: drinking
- c7: reaching behind
- c8: hair and makeup
- c9: talking to passenger

Note: The images in this dataset were created by State Farm in a controlled environment - a truck pulling a car. These "drivers" weren't really driving, so were not putting anyone in danger.

The dataset includes 22,424 labeled training images and 79,726 unlabeled test images. Only the labeled images were used for this project, so that models could be evaluated and ranked.

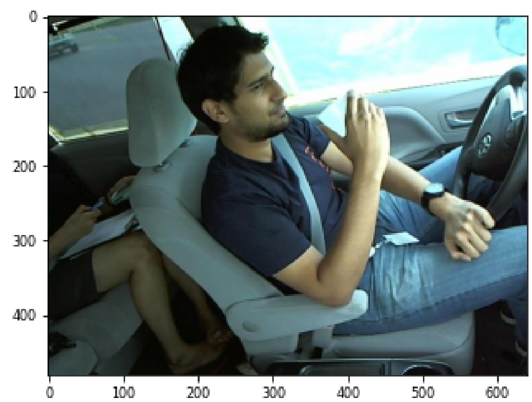
## Exploratory Data Analysis

The image attributes were first checked to determine size and mode distribution. All training images were 640 x 480 pixels, encoded in RGB mode.

Next, images from each category were sampled and pixel intensity histograms were plotted for each color channel. The results varied significantly, depending on the driver's clothes, vehicle interior and seat color, and the amount of light shining into the driver's window.

The cameras in the vehicles appeared to be in slightly different locations. Some showed part of the dash and little to none of the back seat; whereas others showed none of the dash and more of the back seat.

There was a significant amount of extraneous data in many images. The images with the most data unrelated to drivers and distractions came from the cameras showing more of the backseat. Some images included space above drivers' heads, while others displayed the seat cushion at the bottom of the frame.





The 22,424 training images were created with only 26 drivers, all on the same day. The drivers were dressed the same in every image in which they appeared. Each driver was included in every class, although the drivers' images were not distributed evenly across the classes. In addition, the total number of images per driver was not the same, ranging from 346 to 1237.

The 10 classes are close to the same size in the training set, with each class containing between 8.5 and 11.1% of the total images.


```
Total number of images:  22424

Percentage of images per class:
classname
c0    11.099715
c1    10.109704
c2    10.332679
c3    10.462005
c4    10.372815
c5    10.310382
c6    10.368355
c7     8.927934
c8     8.522119
c9     9.494292
dtype: float64
```

# Data Wrangling

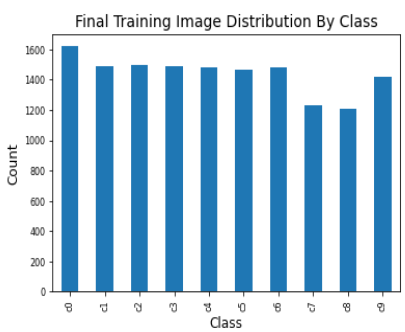
## Train/Validate/Test Split

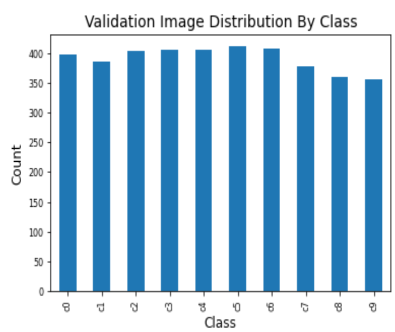
The images were split into three sets: training, validation, and testing. The sets were split by driver to avoid data leakage and ensure results would be generalizable to new drivers.

A pivot table was constructed with counts of images per class for each driver. This table was used with train_test_split() to split off first the test data, then the validation data. The goal was 60/20/20; however, the result was not exact due to splitting by driver.

In the end, the split was 64.2/17.4/18.4, with the image count 14387/3907/4130. This is not many images for training, but a reasonable number were needed for hyperparameter tuning and for model evaluation.

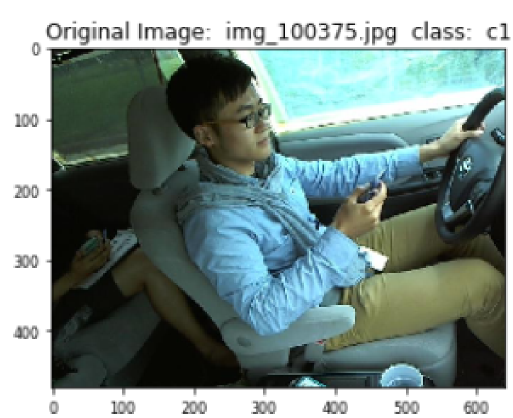
The distribution of each set of images among the classes was similar to the original.

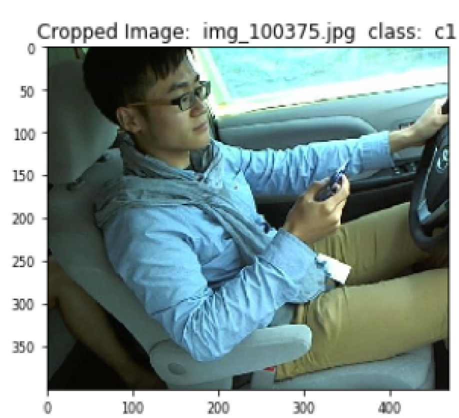





## Cropping

The images were cropped to eliminate as much extraneous data as possible while leaving the driver and distractions intact. Various cropping boxes were tested until one was identified that was a good compromise between cutting enough but not too much. The final crop took 40 pixels off the top, 40 pixels off the bottom, 120 pixels off the left, and 50 pixels off the right of each image.






Once the images were cropped and the train/validation/test sets were defined, the image directory structure for deep learning was built as illustrated below.

```
├── proc     <- The final, canonical data sets for modeling.
│   ├── imgs        <- image data
│   │   ├── train       <- Training images
│   │   │   ├── c0      <- c0 class
│   │   │   ├── c1      <- c1 class
│   │   │   ├── c2      <- c2 class
│   │   │   ├── c3      <- c3 class
│   │   │   ├── c4      <- c4 class
│   │   │   ├── c5      <- c5 class
│   │   │   ├── c6      <- c6 class
│   │   │   ├── c7      <- c7 class
│   │   │   ├── c8      <- c8 class
│   │   │   ├── c9      <- c9 class
│   │   ├── valid       <- Validation images
│   │   │   ├── c0      <- c0 class
│   │   │   ├── c1      <- c1 class
│   │   │   ├── c2      <- c2 class
│   │   │   ├── c3      <- c3 class
│   │   │   ├── c4      <- c4 class
│   │   │   ├── c5      <- c5 class
│   │   │   ├── c6      <- c6 class
│   │   │   ├── c7      <- c7 class
│   │   │   ├── c8      <- c8 class
│   │   │   ├── c9      <- c9 class
│   │   ├── test        <- Testing images
│   │   │   ├── c0      <- c0 class
│   │   │   ├── c1      <- c1 class
```

```
│   │   │   ├── c2     <- c2 class
│   │   │   ├── c3     <- c3 class
│   │   │   ├── c4     <- c4 class
│   │   │   ├── c5     <- c5 class
│   │   │   ├── c6     <- c6 class
│   │   │   ├── c7     <- c7 class
│   │   │   ├── c8     <- c8 class
│   │   │   ├── c9     <- c9 class
```

After the images were placed according to the above diagram, the data structure was processed into a zip file for transporting between platforms.

### Augmentation

Image augmentation was used to improve results due to the small number of training images, and to assist with generalization of models.

Augmentation was done with Keras ImageDataGenerator(). Various options were visualized to determine the largest parameter values that could be used without drivers and distractions being cut out of the images.

The augmentation options used were:

- zoom,
- shear,
- width_shift,
- height_shift, and
- rotation.

Flips were not used due to right/left-handed distractions being different classes.

## Inferences

As this data was staged, no inferences of actual distracted driving can be made. Nevertheless, as the data depicts real distractions, models trained with these images could be used to recognize distracted driving captured by deployed cameras.

## Next Steps

Convolutional Neural Networks will be developed and trained to classify the images according to the defined categories. A simple model will be compared to more complex models and transfer learning to determine which is most accurate.

## References

Distracted Driving. (2019, Sep 16) Retrieved from: https://www.cdc.gov/motorvehiclesafety/distracted_driving/index.html

The Risks of Distracted Driving. (2017) Retrieved from: https://exchange.aaa.com/safety/distracted-driving/the-risks-of-distracted-driving/#.Xsm4jGhKiUk

Distracted Driving. (2020, Feb) Retrieved from: https://www.iihs.org/topics/distracted-driving

National Center for Statistics and Analysis. (2020, April). Distracted driving 2018 (Research Note. Report No. DOT HS 812 926). National Highway Traffic Safety Administration.

Distracted Driving. (n.d.) Retrieved from: https://www.ghsa.org/state-laws/issues/distracted%20driving