kaggle

State Farm Distracted Driver Detection

We've all been there: a light turns green and the car in front of you doesn't budge. Or, a previously unremarkable vehicle suddenly slows and starts swerving from side-to-side. When you pass the offending driver, what do you expect to see? You certainly aren't surprised when you spot a driver who is texting, seemingly enraptured by social media, or in a lively hand-held conversation on their phone.

According to the CDC motor vehicle safety division, one in five car accidents is caused by a distracted driver. Sadly, this translates to **425,000** people injured and **3,000** people killed by distracted driving every year.

Agenda

Project overview

- Project pipeline
- Project complexities
- Solution stages

Neural network

Convelution neural network

Pre-trained model

Drivers Distractions Classes

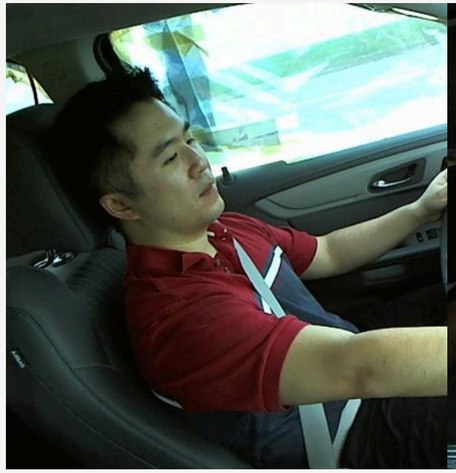
normal driving

texting - right

talking on the phone - right

'texting - left

talking on the phone - left

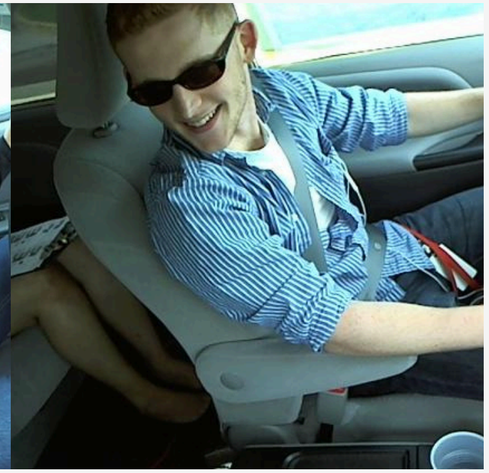


operating the radio

drinking

reaching behind

hair and makeup



talking to passenger

Model Pipeline

- Data Preparation: Collect and preprocess a labeled dataset of images, resizing and normalizing them for consistency.
- Model Design: Choose a neural network architecture suitable for image classification, such as ,NN, CNN, Pretrained model
- Training: Train the model using the prepared dataset, adjusting weights through backpropagation to minimize the loss function.
- Evaluation: Assess the model's performance on a separate validation set to ensure it generalizes well to new images.

Project Complexities

- **Unexpectedly High Accuracies**: Achieving **99%** accuracy on training, validation, and testing sets, which is unusually high and suggests potential **overfitting** or **data leakage**.
- **Class Imbalance:** The model can identify certain classes much better than others, indicating an imbalance or bias in the training data.
- **Network Depth Limitation:** Difficulty in deepening the neural network, possibly due to hardware limitations or vanishing/exploding gradient problems.
- I**mbalanced Dataset Sizes:** The training dataset is significantly smaller than the test dataset, which can lead to poor generalization.
- **Discrepancy in Difficulty:** The difficulty levels of training and testing datasets differ greatly, affecting model performance and reliability.

Unexpectedly High Accuracies

Are they alike??



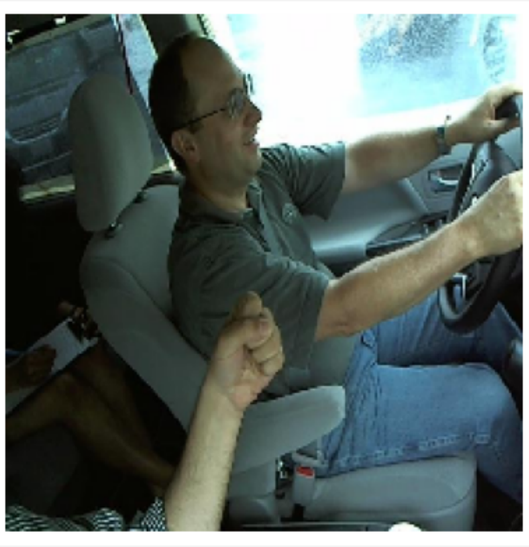


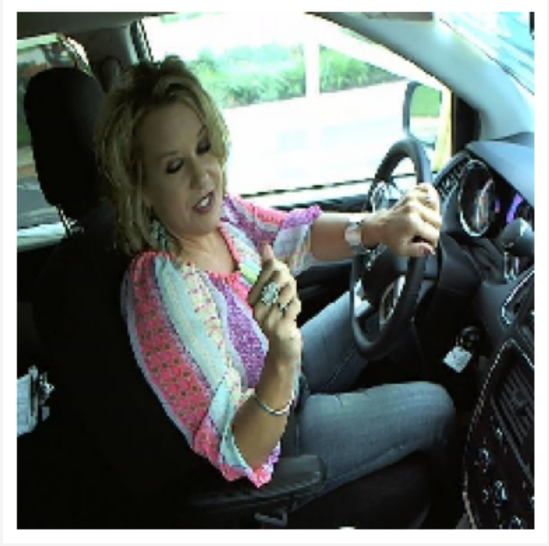

Convert the pictures to drawing lines

talking on the phone - left

Discrepancy in Difficulty

Each picture belongs 2 or more classes



NN Model Design

- **Input Layer:** The number of neurons should match the number of pixels in the flattened image (224x224 image, the input layer will have 50176 neurons).
- **Hidden Layers:** Add one or more hidden layers with a suitable number of neurons (224, 224). Use activation functions like ReLU.
- **Output Layer:** The number of neurons should match the number of classes. Use a softmax activation function for multi-class classification.

CNN Model Design

- **Input Layer:** The input layer will match the dimensions of the input images, which in this case are 224x224 pixels with 3 color channels (RGB).
- **Convolutional Layers:** Add convolutional layers to detect features in the image, followed by pooling layers to reduce spatial dimensions.
- **Fully Connected Layers:** After the convolutional layers, flatten the feature maps and pass them through fully connected (dense) layers.
- **Output Layer:** The output layer should have neurons equal to the number of classes, using a softmax activation function for multi-class classification.

CNN Model Architecture

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 223, 223, 64)	832
max_pooling2d (MaxPooling2D)	(None, 111, 111, 64)	0
conv2d_1 (Conv2D)	(None, 110, 110, 128)	32,896
max_pooling2d_1 (MaxPooling2D)	(None, 55, 55, 128)	0
conv2d_2 (Conv2D)	(None, 54, 54, 256)	131,328
max_pooling2d_2 (MaxPooling2D)	(None, 27, 27, 256)	0
conv2d_3 (Conv2D)	(None, 26, 26, 512)	524,800
max_pooling2d_3 (MaxPooling2D)	(None, 13, 13, 512)	0
dropout (Dropout)	(None, 13, 13, 512)	0
flatten (Flatten)	(None, 86528)	0
dense (Dense)	(None, 500)	43,264,500
dropout_1 (Dropout)	(None, 500)	0
dense_1 (Dense)	(None, 10)	5,010

Pre-trained Model

- **Pre-trained Base Model:** The model is used as the base, excluding its top fully connected layers. This base model will provide feature extraction capabilities with weights pre-trained on ImageNet.
- **GlobalAveragePooling2D Layer:** This layer reduces the spatial dimensions of the feature maps from the base model, effectively summarizing the features.
- **Dense Layers:** These are custom fully connected layers added on top of the base model. The example uses two Dense layers with 224 neurons each and ReLU activation.
- **Output Layer:** The final Dense layer has neurons equal to the number of classes (10) and uses a softmax activation function to output probabilities for each class.

Thanks