Distracted Driver Detection

Domain Background

Distracted driving is a growing public safety hazard. The dramatic rise in texting and constant updates from social media platforms appear to be contributing to an alarming increase in distracted driving fatalities. It accounts for approximately 25% of all motor vehicle crash fatalities. These deaths all stem from a cause that is entirely preventable. Legislation enacting distracted driving bans should be paired with effective enforcement to dissuade drivers from distractions while driving. Providing tools to alert users when they are distracted can make the user self-aware of his/her actions and notice the frequency at which they are getting distracted while driving so that they can take steps to reduce or remove the sources of distraction.

The purpose of this project is to classify a dataset of images of drivers at the wheel to identify distracted drivers and categorize them based on the type of distraction using Computer vision. Computer Vision is a branch of Machine Learning that is concerned with the automatic extraction, analysis, and understanding of useful information from a single image or a sequence of images. Today's Smart Cars use sensors, Lidar (Light Detection and Ranging), radar, cameras, and image recognition systems to "see" the environment around them. Features can range from autonomous driving to safety features such as emergency braking and animal detection through the use of radar. Computer Vision will change the way we commute in the future and shape the consumer goods delivery sector.

This problem interests me because I want to specialize in Self Driving Cars, and I see a project in Computer Vision as the right first step in that direction.

Problem

The goal of this project is to build a model to classify an image of a driver at the wheel as distracted or not distracted and if the driver is distracted, name the type of distraction. The model will be trained using a dataset of images of drivers taken using a dashboard camera. The dataset used, "State Farm Distracted Driver Detection," is provided by State Farm and it is published on the Kaggle website.

Each image in the dataset is classified into one of the following ten classes.
0: safe driving
1: texting - right
2: talking on the phone - right
3: texting - left
4: talking on the phone - left
5: operating the radio

6: drinking
7: reaching behind
8: hair and makeup
9: talking to a passenger

So the problem to be solved is to predict the likelihood of what the driver is doing in each picture. For each image in the test set, the probability of the image belonging to each of the classes is calculated. Since images in the test set are labeled, each image's right class can be compared with calculated probabilities to quantify and measure the trained model.

Datasets and Inputs

State Farm published the dataset used here on the Kaggle website as part of State Farm Distracted Driver competition. It is a dataset of JPEG images of size 649* 480. The driver images are taken using a dashboard camera in a car with a driver doing something in the car such as texting, eating, talking on the phone, makeup and reaching behind.
The dataset is divided into two folders train and test data with approximately 25000 images for training and 102,000 images for testing. The train and test data are split such that once driver can only appear on either test or train set. I will use cross-validation on the training set to fine tune the model and use test data to evaluate the performance of the model.

Solution Statement

For this multi-class classification problem of images, the apparent solution was to use Convolutional Neural Networks(CNN), which captures spatial characteristics of images and transform them into hierarchical invariant features for classification. The final model will be a weighted combination of an ensemble of CNN classifiers. Each of these CNNs will be trained to classify on input images that are cropped to focus on specific features such as the face, hands, eyes, steering wheel or certain parts of the image, like, the right half of the image which has information on if the hands are on the steering wheel. Inputs to each of these models and the number of models will be decided during the model development phase depending on trained models publicly available to detect body parts, subject, and objects in the image relevant to this use case. The classification response from each of these networks is combined with a CNN trained on the entire image to form a final classification response.

Various architectures like AlexNet, ZFNet, VGGNet, GoogLeNet, ResNet have established benchmarks in computer vision. In this project, using the technique of transfer learning, I will use one or more of these models as feature encoders, with an additional classification layer. Transfer learning will reduce training time while still producing good results.

The test set is much bigger than the train set, so it is easy to get overfit. In order to be able to train a model with enough data, I will use data augmentation to increase generalization of the training set and increase accuracy on the test set.

Benchmark Model

VGG16 will be used as the benchmark model. A pre-trained VGG16 model will be used as a fixed feature extractor, with an additional fully-connected softmax layer with ten outputs for classification.

Evaluation metrics

Cross-entropy, or log loss, will be used to measure the performance of the model. For each image in the test, predicted probabilities will be calculated using the final model. The images in the test set are labeled with a true class

A cost function based on multiclass log loss for a dataset of size N will look like this:

$$F = -\frac{1}{N}\sum_{i}^{N}\sum_{j}^{M} y_{ij} \cdot Ln(p_{ij})) = \sum_{j}^{M}\left(-\frac{1}{N}\sum_{i}^{N} y_{ij} \cdot Ln(p_{ij}))\right) = \sum_{j}^{M} F_i$$

where N is the number of instances, M is the number of different labels, yij is the binary variable with the expected labels and pij is the classification probability output by the classifier for the i-instance and the j-label.

Project Design

1. Data loading and visualization: Load the data and get familiar with the data and distribution amongst categories using visualizations.
2. Create a CNN without transfer learning and train it, this CNN will provide a lower performance bound.
3. Create a benchmark model using pre-trained VGG16 by adding a fully-connected softmax layer with ten outputs for classification.
4. Train models using AlexNet, Inception and ResNet available and choose the model that performs best on the training set as the feature encoder.
5. Enhance the results of the model chosen using hyperparameter tuning and data augmentation.
6. Use publicly available libraries to identify specific areas of the images that can be trained separately and combined with the model built in the previous step to enhance the performance of the model.
7. Once the segmentation is decided, using pre-trained models available as feature extractors

train separate models with output images from segmentations performed in the previous step.
8. Combine the CNNs trained independently in specific areas with the CNN model trained on the entire image to create the final model which is a weighted combination of the results from all the CNN models.

This project will be implemented in Python using Tensorflow and Keras libraries.

References

1. Trends in Fatalities From Distracted Driving in the United States. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2951952/
2. https://www.teensafe.com/distracted-driving/100-distracted-driving-facts-and-statistics-2018/
3. Computer Vision. http://www.bmva.org/visionoverview
4. https://www.artimus-uk.com/computer-vision/
5. https://www.kaggle.com/c/state-farm-distracted-driver-detection
6.https://jamesmccaffrey.wordpress.com/2013/11/05/why-you-should-use-cross-entropy-error-instead-of-classification-error-or-mean-squared-error-for-neural-network-classifier-training/
7.https://stats.stackexchange.com/questions/113301/multi-class-logarithmic-loss-function-per-class
8. Distracted driving is a growing public safety hazard .... https://drivingmba.com/distracted-driving-is-a-growing-public-safety-hazard/
9. Multi-class logarithmic loss function per class. https://stats.stackexchange.com/questions/113301/multi-class-logarithmic-loss-function-per-class
10. ARTIMUS Ξ Artificial Intelligence Company & Machine Learning. https://www.artimus-uk.com/