AI Based Computer Vision Techniques for Distracted Drivers Detection

Introduction

Distracted driving is a main factor that cause severe car accidents. It has been suggested as a possible contributor to the increase in fatal crashes from 2014 to 2018 and is a source of growing public concern[1]. This project focuses on **driver distraction activities detection** via images, which is useful for vehicle accident precaution. We aim to build a high-accuracy classifiers to distinguish whether drivers is driving safely or experiencing a type of distraction activity.

Dataset

The training dataset contains **22,424 images** categorized in **10 classes** from *State Farm*®. We randomly split the dataset into two folds: **80% for training**, **20% for validation**. One category represents safety driving, and other 9 categories represents 9 different distraction activities we consider here.

Figure 1: Examples of different classes in dataset

Images in the dataset have very high resolutions $(480 \times 360 \times 3)$, and in order to improve the computational efficiency, we preprocessed the images by resizing them to $(64 \times 64 \times 3)$, followed by flattening the high dimensional image matrix to image vectors as the input to train the classifiers.

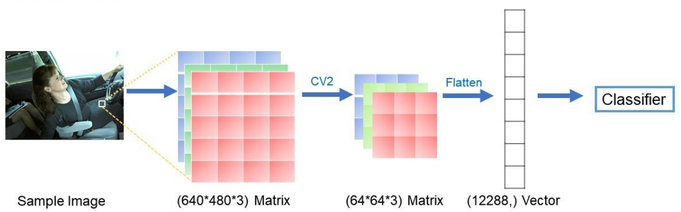


Figure 2: Float Chart of Image Processing

Models

- **Linear Support Vector Machine (SVM) Classifier:**

$$L = \frac{1}{N}\sum_{i=1}^{N}\sum_{j\neq y_i}\max(0, f(x_i,W)_j - f(x_i,W)_{y_i} + 1) + \lambda||W||_2^2$$

- **Softmax Classifier:**

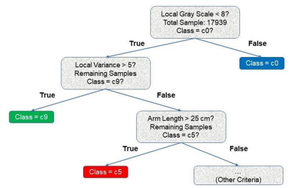
$$L = \frac{1}{N}\sum_{i=1}^{N} -\log(\frac{\exp f(x_i,W)_{y_i}}{\sum_j \exp f(x_i,W)_j}) + \lambda||W||_2^2$$

- **Naïve Bayes Classifier :**

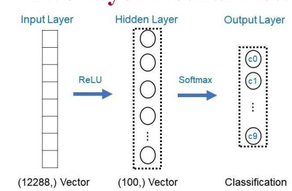
$$P(x_i \mid c_k) = \frac{1}{\sqrt{2\pi\sigma_{c_k}^2}}\exp\left(-\frac{(x_i-\mu_{c_k})^2}{2\sigma_{c_k}^2}\right)$$

$$\hat{c_k} = \underset{c_k}{\operatorname{argmax}}\, P(c_k)\prod_{i=0}^{n} P(x_i \mid c_k)$$

- **Decision Tree Classifier:**



- **Two-layer Neural Net:**



Results

The **two-layer neural net** model gives the best validation set accuracy of **92.2%**, which meets our expectation that CNN-based models will have better performance on computer vision task than other modes[2].

Classifiers	Training Accuracy	Evaluation Accuracy
Naïve Bayes	N/A	54.99
Decision Tree	N/A	84.73
Linear SVM	72.82	71.39
Softmax	82.32	82.31
Two Layer Net	93.21	**92.24**

Table 1: Models Evaluations

For our **best model**, we studied the **per-class** accuracy and found out that compared with other class, the model has lowest accuracies on **"talking on the phone - left"** and **"hair and makeup"** class, which are below **80%.**

Class	Accuracy	Class	Accuracy
Safe Driving	89.20	**Operating the Radio**	98.28
Texting – Right	97.80	**Drinking**	98.03
Talking on the Phone – Right	95.83	**Reaching Behind**	97.08
Texting – Left	98.37	**Hair and Makeup**	**78.97**
Talking on the Phone – Left	**71.24**	**Talking to Passenger**	96.20

Table 2: Per Class Accuracy on Two Layer Net

Conclusion

- After stabilizing the randomness, improving the weight initialization and redoing the hyperparameters tuning of the SVM classifier, the accuracy increased from ~55% to 71%.
- Naïve Bayes is not a good choice for image classification tasks.
- The overall accuracy of the rest models are high, even SVM and decision tree classifiers can achieve 70% ~ 80%, and we think this is because the features of the images are not very complicated.
- For Two-layer Neural Net, it sometimes predicts the "Talking on the Phone-Left" to be "Texting-Left" due to the similarities of these two classes and leads to the relatively low accuracy on this class.
- CNN are still the **state-of-the-art** models for computer vision tasks.

Future Work

- For SVM, Softmax and Two Layer Net, we could continue tune other hyperparameters or use some more mature weight initialization techniques like Xavier Initialization or KaiMing Initialization.
- Some other techniques such as Logistic Regression, K-Nearest Neighbors or Random Forest can give sense of how well traditional machine learning techniques can perform on computer vision.
- Instead of implementing by ourselves, we could utilize the existing packages of more advanced CNN-based models like ResNet or VGG.
- The dataset also contains 77,000 testing images without providing labels. We could apply these samples to evaluate our classifiers better if these images are labelled or via semi-supervised learning method.