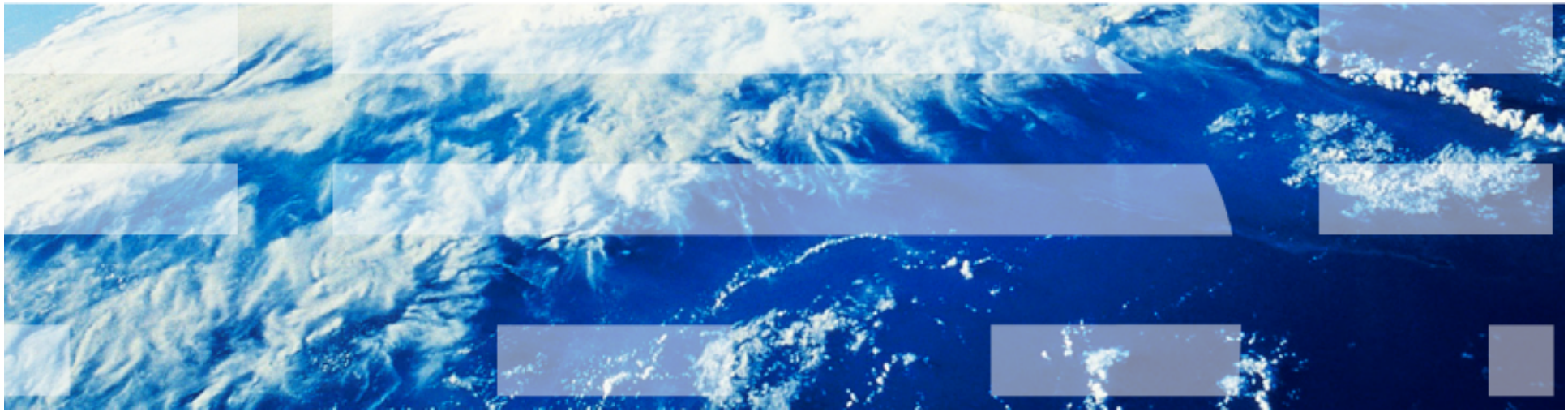
COLUMBIA UNIVERSITY

E6893 Big Data Analytics:

Drive and Arrive – Distracted Driver Detection

Team Members: Jiaming Liu (jl4564, Management Science and Engineering)
Yuhang Liu (yl3380 , Management Science and Engineering)
Yaqing Xie (yx2316 , Management Science and Engineering)



December 15, 2016

Outline

COLUMBIA UNIVERSITY

1. Motivation and Dataset

2. Face Recognition

3. Feature Extraction
- Tool: PySpark
- Caffe
- OpenCV

4. Feature Decomposition
- Tool: PySpark
- Principal component analysis

5. Model Training and Selection

6. Interface and Demo
- Tool: R Shiny

7. Next Steps

Motivation and Dataset

Background

- 1/5 car accidents are caused by distracted driving
- 425,000 people injured and 3,000 people killed every year

Objective

Enable dashboard cameras to automatically detect drivers engaging in distracted behaviors to avoid traffic accidents.

Dataset

The dataset was collected from a Kaggle competition: State Farm Distracted Driver Detection. It includes two files: imgs.zip, a zipped folder of all (train/test) images, and driver_imgs_list.csv, a list of training images, their subject (driver) id, and class id.

Face Recognition

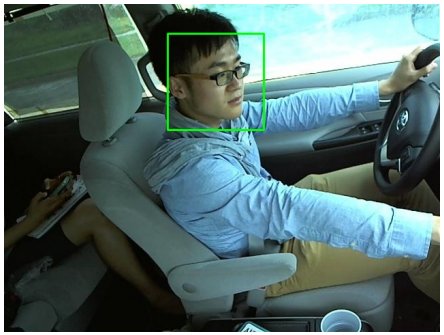
Logic Behind

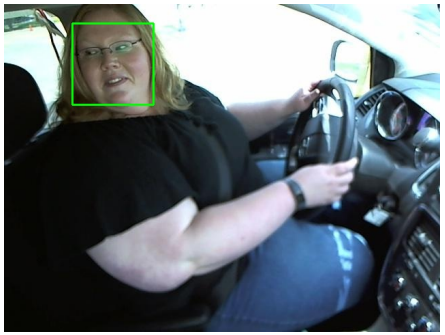
As our system to detect distracted driving behavior should definitely be real-time, it's essential to reduce computing time and save computing resources. Thus we first did face recognition, only do we identify the driver's face will we start to detect distracted driving behavior.

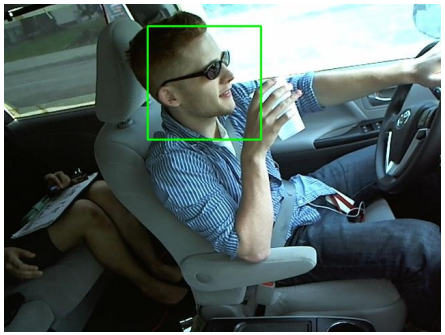
Procedures

- Load default trained face Haar-cascade detection model
- Identify whether there's a driver's face in the pic
- Locate the face and use rectangle to point it out

Some examples:





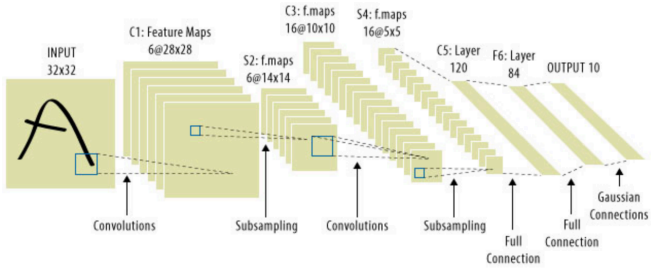


Feature Extraction

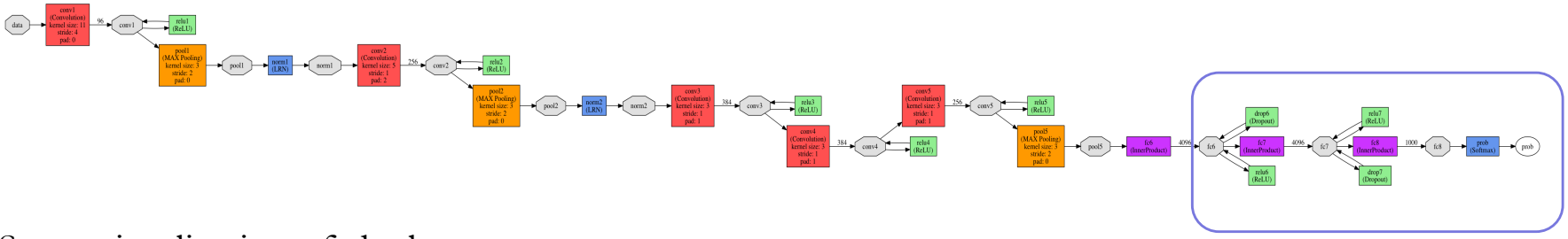
Tool - PySpark

Caffe

Caffe is a deep learning framework made with expression, speed, and modularity in mind. It is developed by the Berkeley Vision and Learning Center (BVLC) and by community contributors.



We adopted the neural network structure below, and chose layer 7/8/9 as candidate features for model training.



Some visualization of the layers:

Feature Extraction

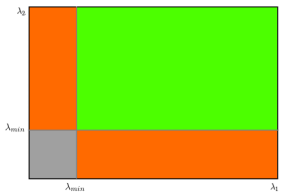
OpenCV

OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision. It was designed for computational efficiency and with a strong focus on real-time applications.

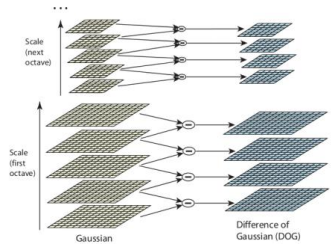
SIFT: Scale-Invariant Feature Transform

We used openCV to extract SIFT features from original images.

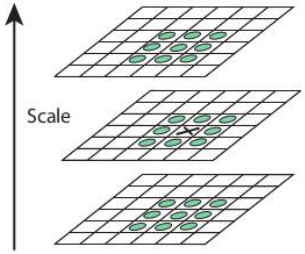
Shi-Tomasi Corner



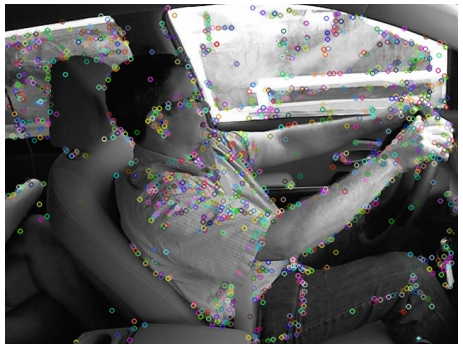
Difference of Gaussians

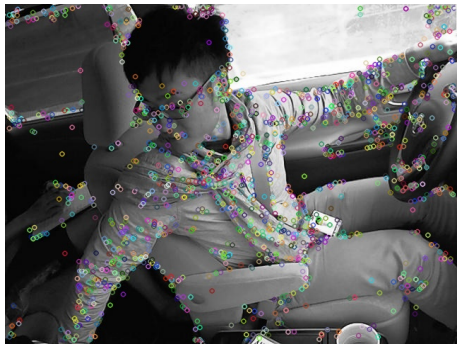


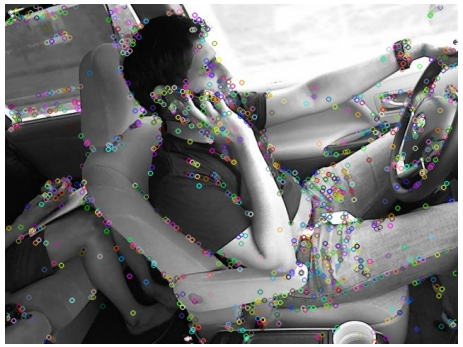
Key Points: local extrema



Some visualization of the SIFT key points:







Feature Decomposition

Tool - PySpark

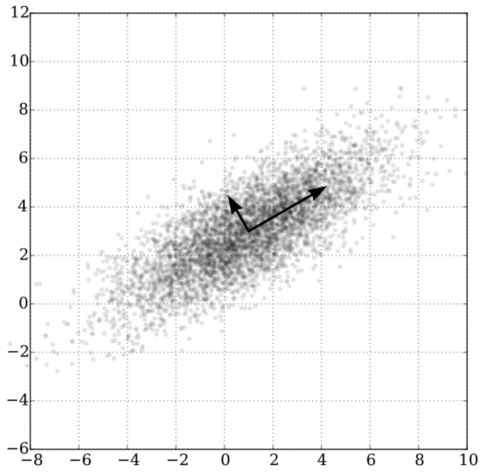
Principal Component Analysis

Methodology:

PCA is a statistical procedure that uses an **orthogonal transformation** to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables. This transformation is defined in such a way that the first principal component has the largest possible variance (that is, accounts for as much of the variability in the data as possible), and each succeeding component in turn has the highest variance possible under the constraint that it is orthogonal to the preceding components.

Dimensions:

Layer 7: 4096 -> 750 (remain 95.41% variance)
Layer 8: 1000 -> 70 (remain 95.77% variance)
Layer 9: 1000 -> 5 (remain 96.15% variance)



Model

Input/Output

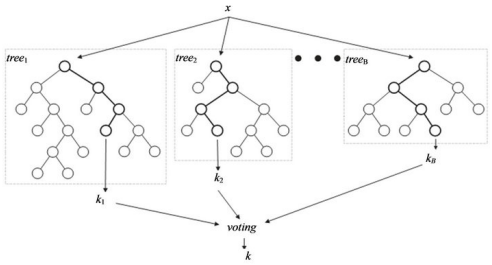
Input : image id with Caffe features.
Output : image id with class label.

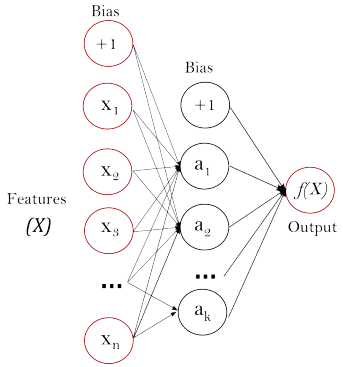
Model Algorithms

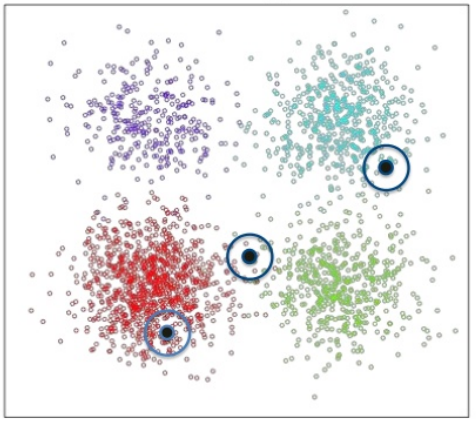
Random Forest: It is operated by constructing a multitude of decision tree at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set.

K Nearest Neighbor: An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors. If k = 1, then the object is simply assigned to the class of that single nearest neighbor.

Neural Network: Multi-layer Perceptron (MLP) is a supervised learning algorithm that learns a function $f(\cdot) : R^m \rightarrow R^o$ by training on a dataset, where m is the number of dimensions for input and o is the number of dimensions for output.







Model

Cross Validation

- Random forest: number of estimators
- K nearest neighbor: n of neighbors

Algorithm/n	3	5	10
KNN	33%	41%	58%
RF	31%	43%	63%

Fine-tuning

	Neural Network	K Nearest Neighbor k = 5	Random Forest n = 10
feature7	54%	69%	**72%**
feature7_pca	65%	69%	66%
feature8	14%	67%	69%
feature8_pca	56%	67%	67%
feature9	21%	41%	64%
feature9_pca	12%	19%	25%

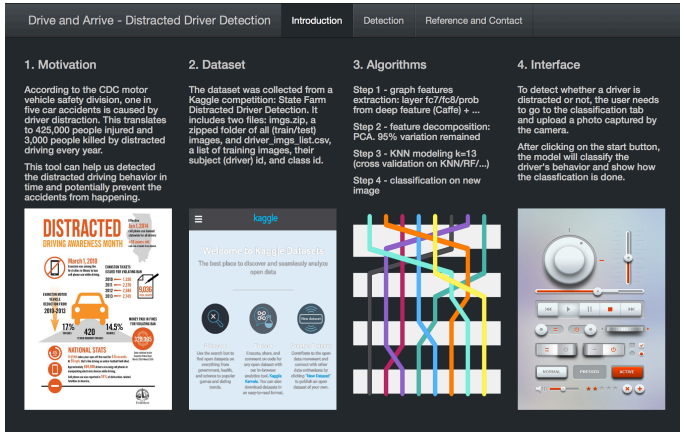
Interface and Demo

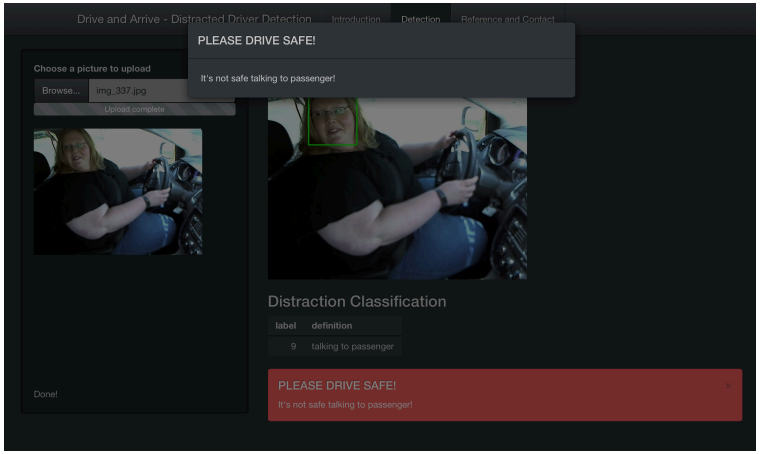
Tool – R Shiny

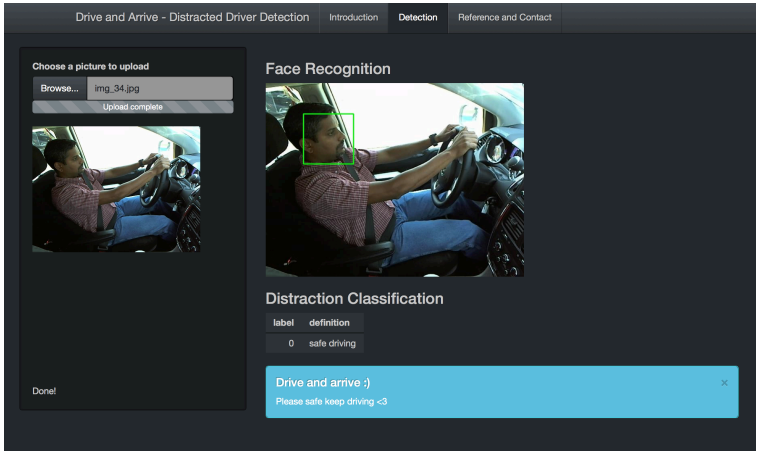
Operation and Guidance

The **Instruction** tab and **Contact** tab give detailed information about this project.
To detect driver distraction, go to **Detection** tab and upload a photo captured by the camera device.

Screenshots







Next Steps

COLUMBIA UNIVERSITY

Problems We Met

- Detection accuracy relies heavily on trained driver data
- Algorithm efficiency to be improved for real-time situation

Next Steps

- Explore new features and new models to apply our system to more realistic situation
- Instead of images, upgrade our system which should be applied to video stream

Q&A

Columbia University

Thank You!