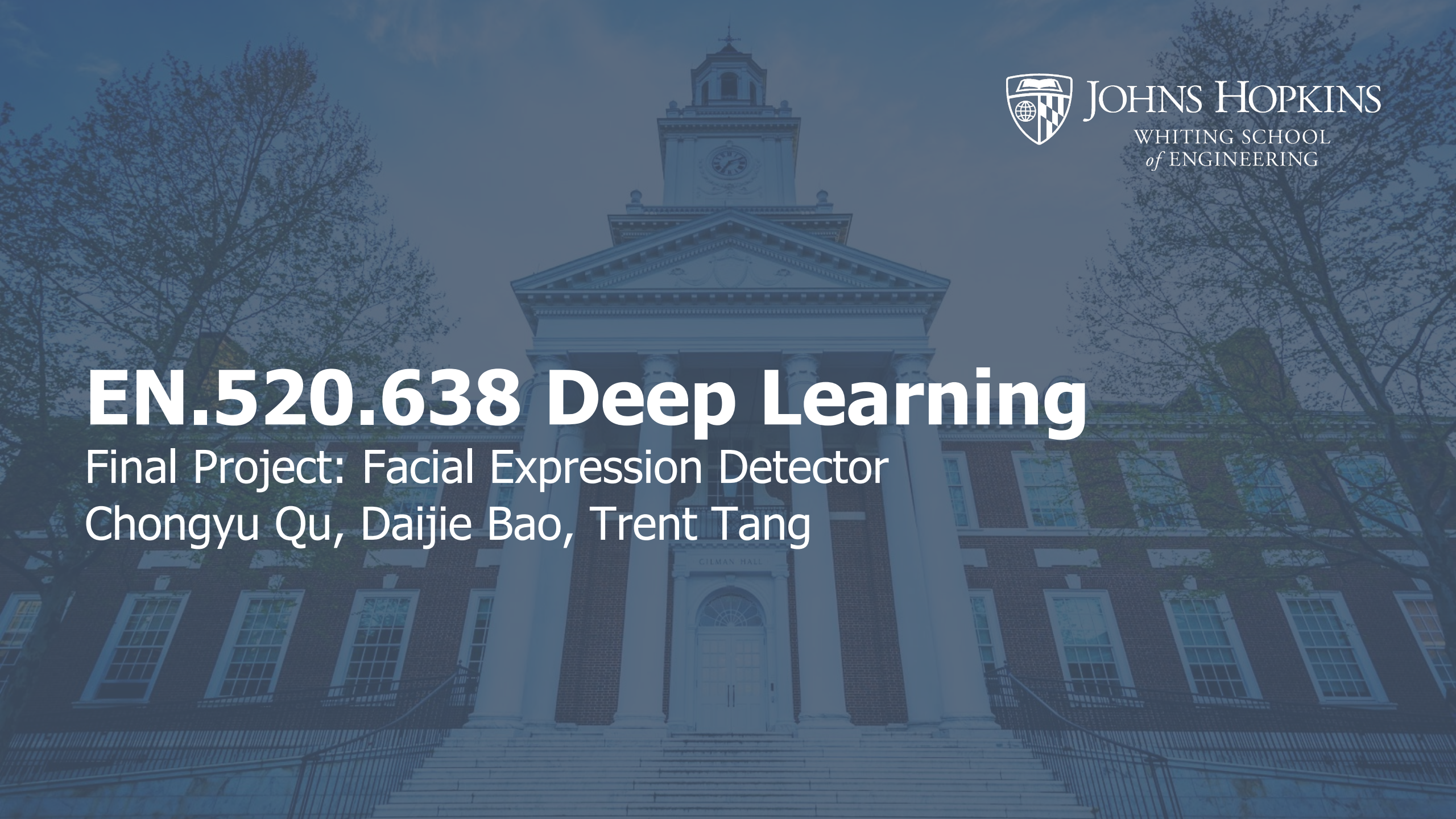


JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

EN.520.638 Deep Learning

Final Project: Facial Expression Detector

Chongyu Qu, Daijie Bao, Trent Tang

Background

Mental Health

- Mental illnesses are an increasingly serious problem in the world. Nearly one in five U.S. adults live with a mental illness.
- Patients with a number of mental illnesses may display abnormal facial expressions.

Old-age Healthcare Management

- There were 703 million persons aged 65 years or over in the world in 2019. The number of older persons is projected to double to 1.5 billion in 2050.
- Face expressions can be used to keep track of individual emotions for health care.

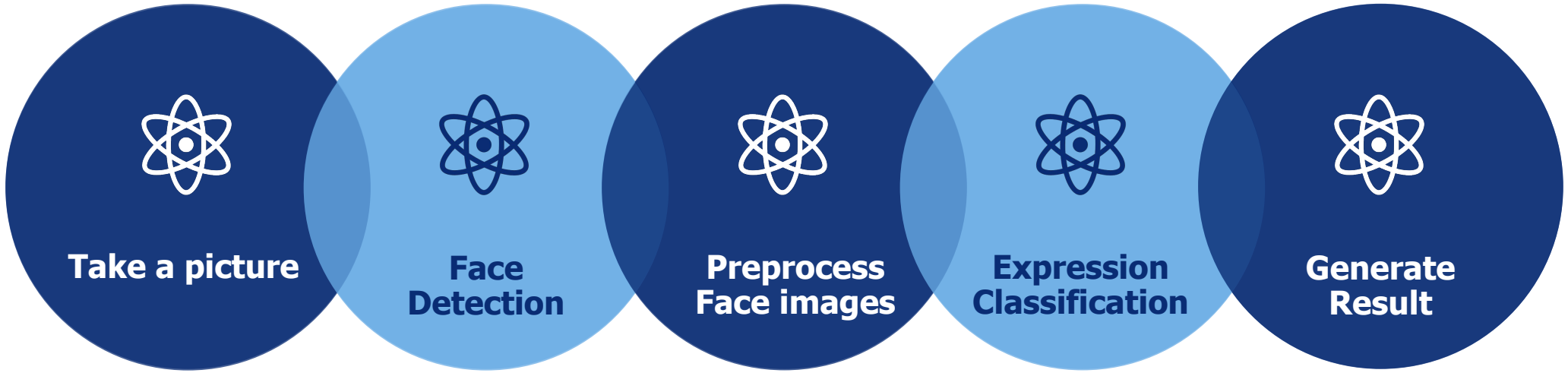
Objective

- Build and train a convolutional neural network that can effectively classify various facial expressions
- Implement traditional computer vision methods like or CNN like MTCNN to detect human faces in the pictures.

Model Pipeline

Use Haar Cascade Classifier or Faster R-CNN to generate bounding boxes around faces

Use VGG16 to classify 8 different facial expressions



Extract and preprocess detected face images for expression classification networks

Label faces of interest and store them for future healthcare management

Datasets

Facial Expression

- AffectNet
- Contains more than 1M labeled facial images collected from the Internet by querying three major search engines using 1250 emotion related keywords in six different languages.

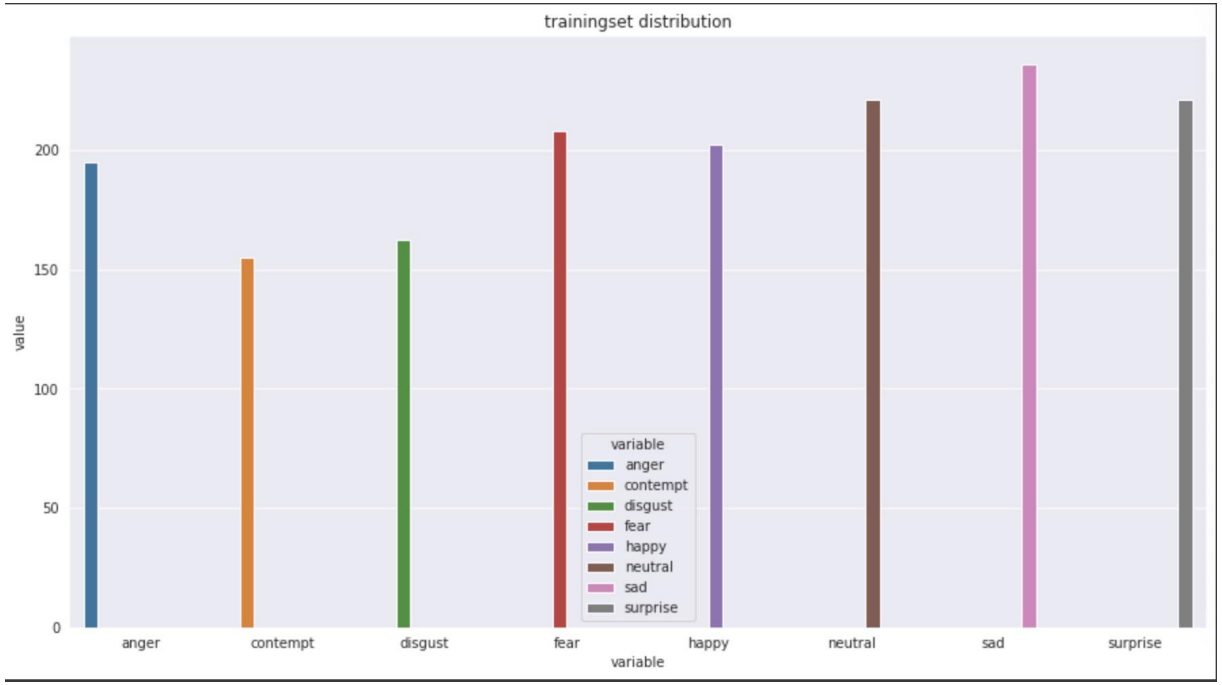
Face Detection

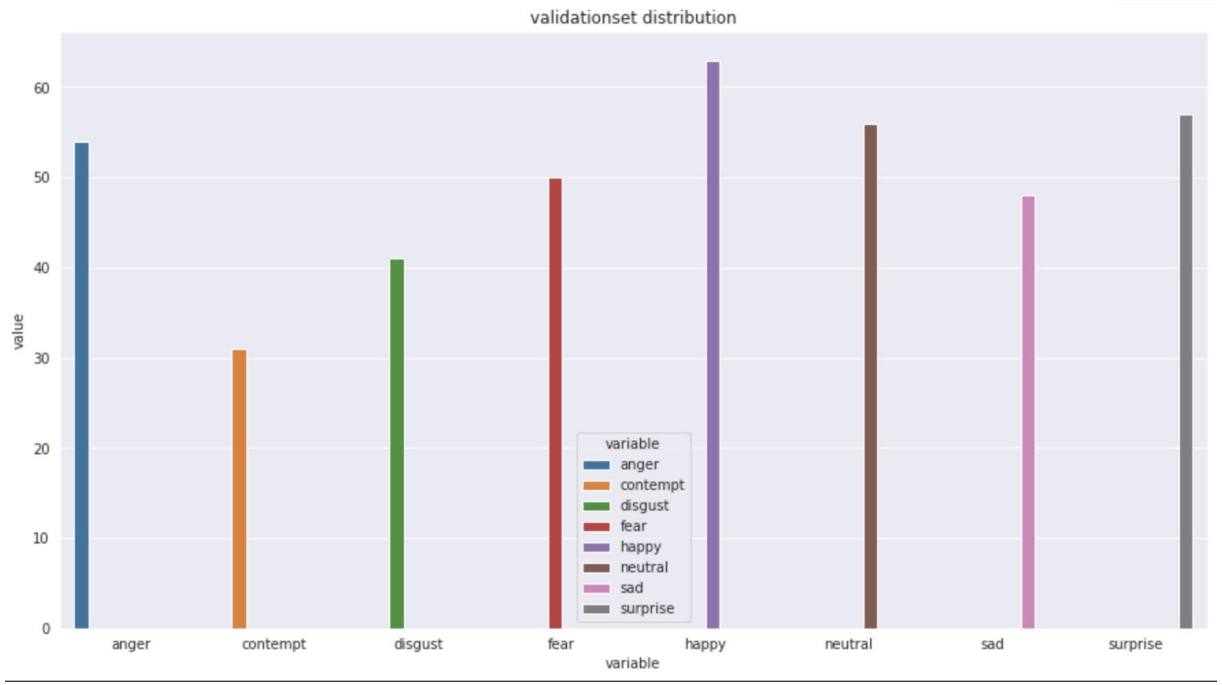
- WIDER FACE
- WIDER FACE dataset is a face detection benchmark dataset, of which images are selected from the publicly available WIDER dataset. They choose **32,203** images and label **393,703** faces with a high degree of variability in scale, pose and occlusion as depicted in the sample images.

Reference: http://mohammadmahoor.com/affectnet/
http://shuoyang1213.me/WIDERFACE/

Datasets

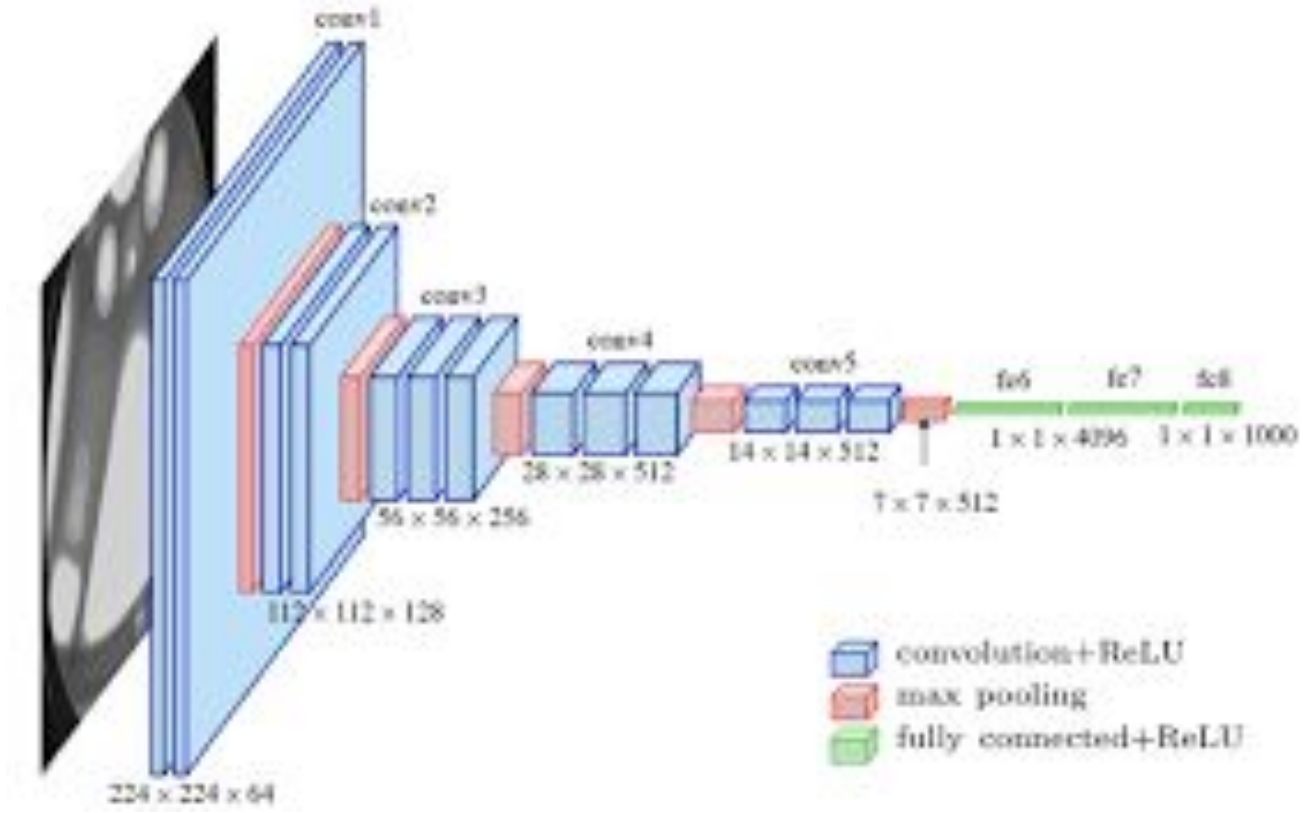
Subset of AffectNet, 1600 training, 400 validation, 5000 test labeled image.
8 different classes.
Each sample is resized to 224*224 to feed to pre-trained models.





Expression Classifier

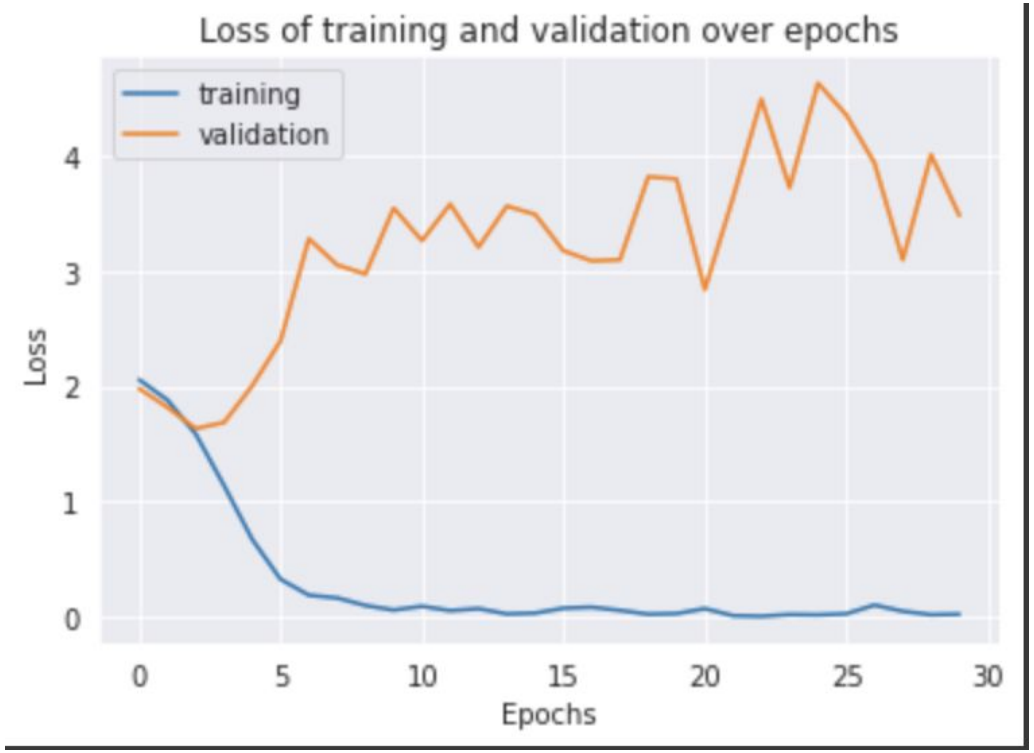
Pre-trained VGG-16: 13 convolutional layers, 5 max pooling layers, 3 fully connected layers.
Fine-tuning on our dataset



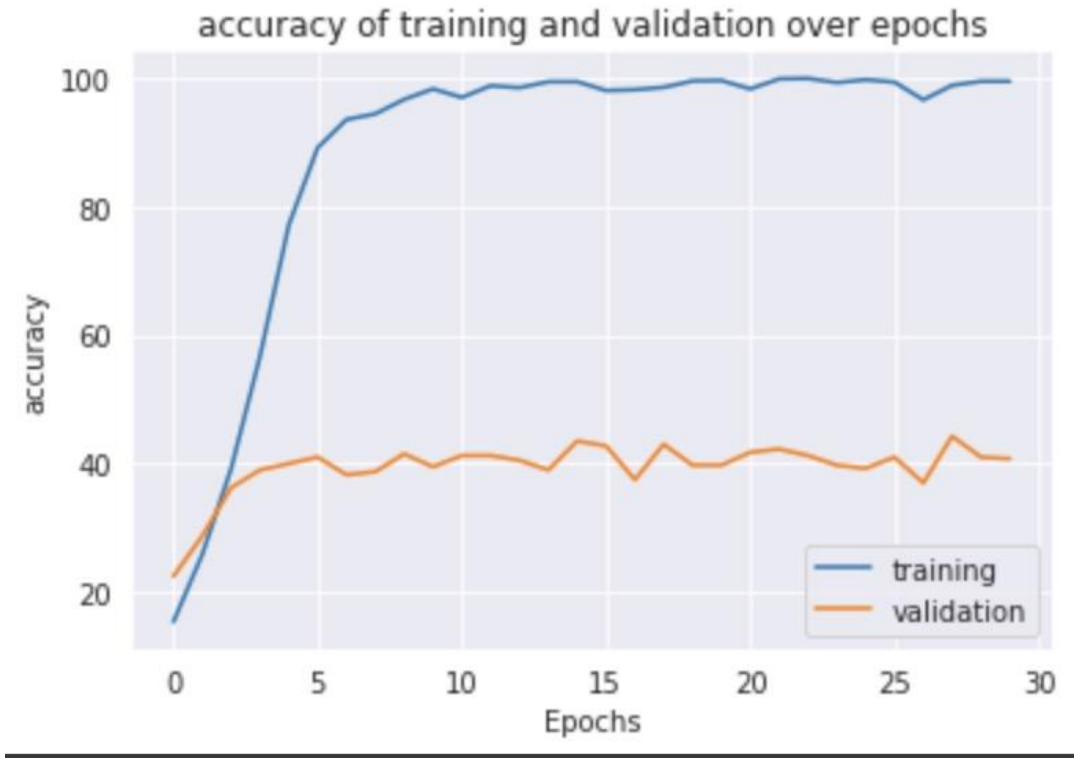
Results Analysis

Train Accuracy: 98.87%
Validation Accuracy: 44.75%

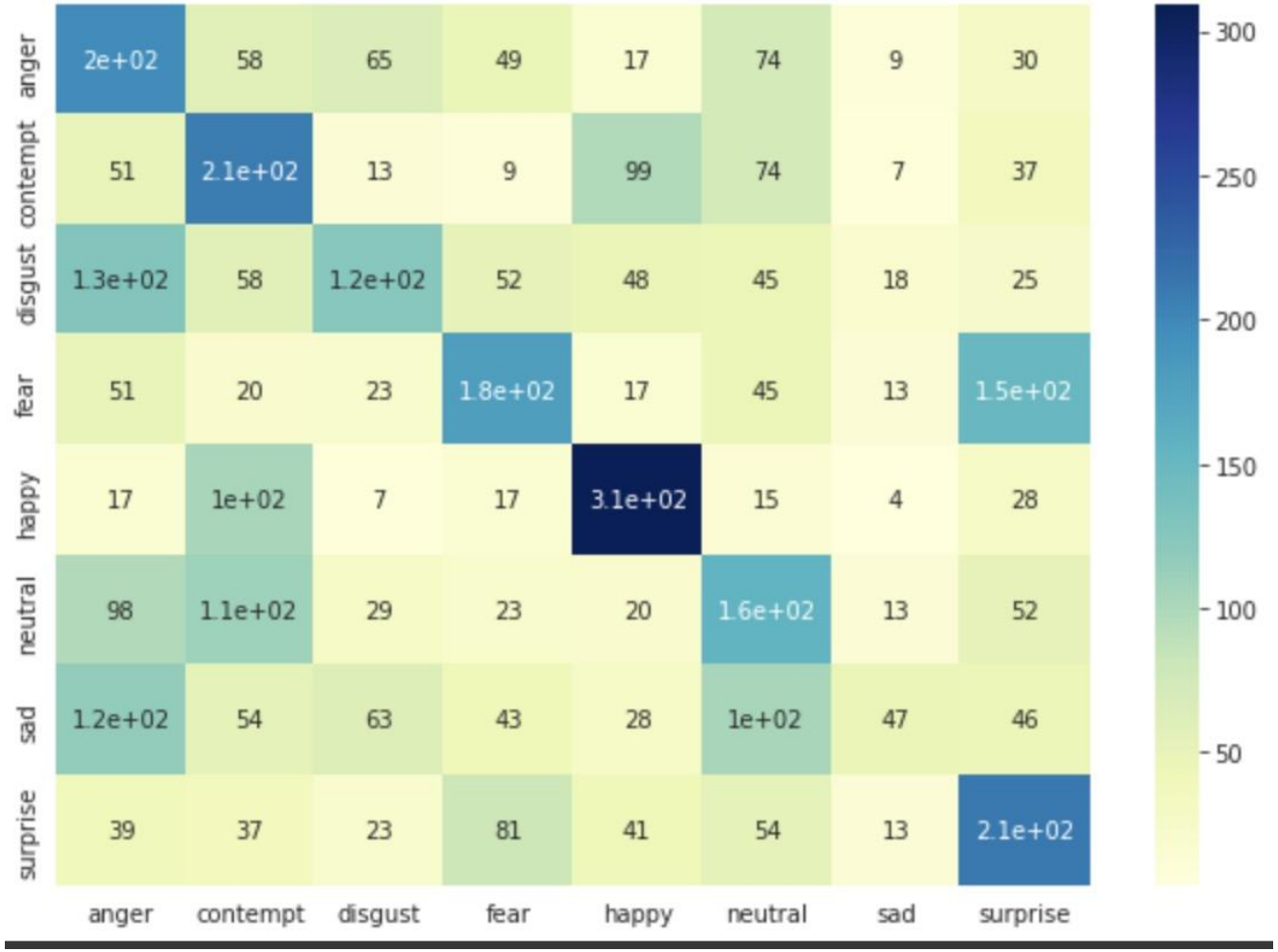
- training and validation loss



training and validation accuracy



Confusion Matrix



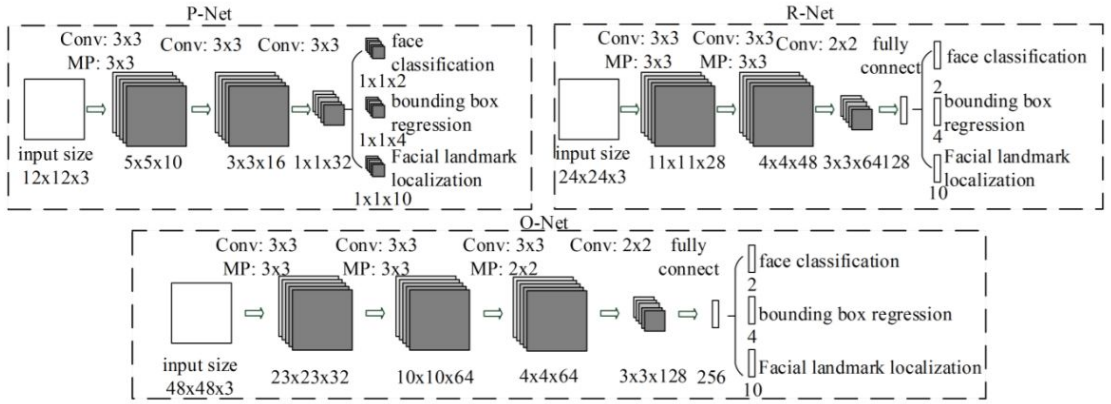
predict on 5000 test images

highest accuracy: happy

lowest accuracy: sad

Face Detector

- Multi-task Cascaded Convolutional Networks (MTCNN)
  - Consists of three stages of convolutional networks that are able to recognize faces and landmark location such as eyes, nose, and mouth.
  - Stage 1: A fully convolutional network (FCN) used to obtain candidate windows and their bounding box regression vectors.(P-Net)
  - Stage 2: The Refine Network (R-Net) reduces the number of candidates, performs calibration with bounding box regression and employs non-maximum suppression (NMS)
  - Stage 3: Output Network aims to describe the face in more detail(O-Net)



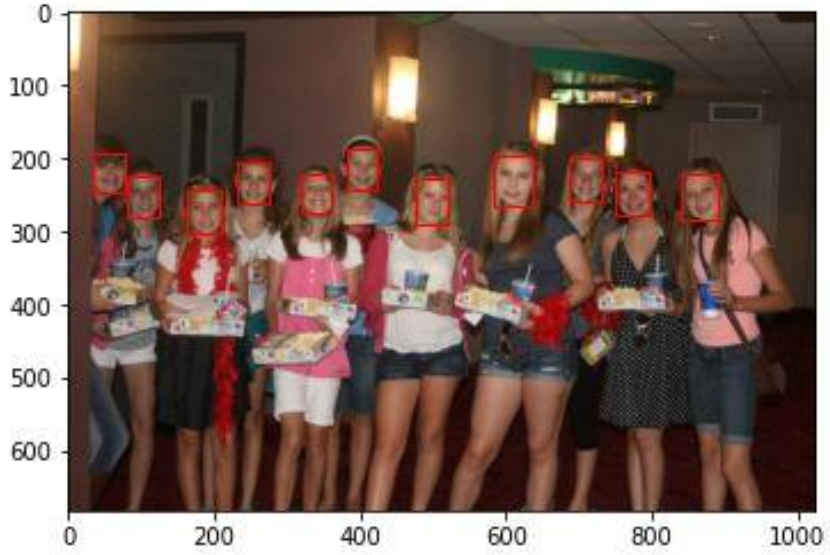
Reference: https://medium.com/@iselagradilla94/multi-task-cascaded-convolutional-networks-mtcnn-for-face-detection-and-facial-landma

Analysis

Single Waitress

Detection coordinates Result for Single Person :
Bounding Box: [359, 112, 165, 218]
Detection Confidence Score: 0.9997734427452087
Face Keypoint: {'left_eye': (411, 200), 'right_eye': (489, 177), 'nose': (472, 236), 'mouth_left': (423, 272), 'mouth_right': (507, 251)}

Birthday Party



Detection coordinates of bounding box:
box: [381, 188, 43, 54], box: [164, 242, 46, 62], box: [81, 223, 44, 57], box: [751, 218, 47, 60], box: [584, 195, 52, 70], box: [233, 207, 43, 56], box: [688, 200, 42, 59], box: [841, 221, 51, 64], box: [477, 228, 45, 61], box: [33, 193, 45, 53], box: [320, 220, 41, 55]

Demo

Real Life image from our group

original image

face detection

cropped and resized image

predicted label: ‘happy’

Future Steps

- Start to improve our model pipeline performance by using different real life test images.
- Further improve our classification model performance by using different fine-tune methods.
- Using regularization methods to improve our model performance.

Thank You

JOHNS HOPKINS

WHITING SCHOOL *of* ENGINEERING