Face Detection And Facial Expression Recognition System

Krupali Mewada(201501056)
Maharsh Suryawala(1641029)
Nikhil Balwani(1641070)
Parshwa Shah(1641068)

Introduction

- Facial expression recognition software is a technology which uses biometric markers to detect emotions in human faces. More precisely, this technology is a sentiment analysis tool and is able to automatically detect the six basic or universal expressions: happiness, sadness, anger, neutral, surprise, fear, and disgust.
- Facial expressions and other gestures convey nonverbal communication cues that play an important role in interpersonal relations.
- Therefore, facial expression recognition, because it extracts and analyzes information from an image or video feed, it is able to deliver unfiltered, unbiased emotional responses as data.

Problem Statement

- Given a data set consisting of facial images and their sketches, retrieve all images (real and /or sketch) which are similar to the given test image, along with various attributes of the image such as gender, expression and so on.

How does facial expression recognition work?

- Face Detection - Locating faces in the scene, in an image or video footage.
- Facial Landmark Detection - Extracting information about facial features from the detected faces.
- Facial Expression And Emotion Classification - Classifying the obtained information into expression interpretative categories such as smile or frown or emotion categories such as happy, anger, disgust etc.

GANTT Chart

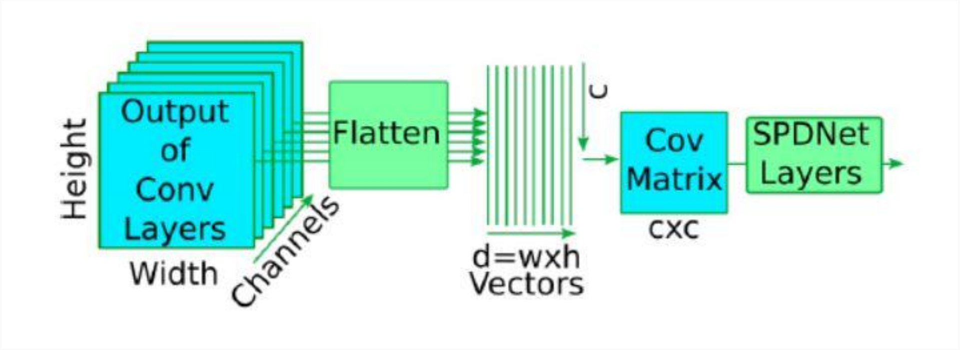
	Start Date	End Date	Timeline	Status
Project Name	Mar 1, 2019	Apr 19, 2019		
Literature Review	Mar 1, 2019	Mar 10, 2019		Complete
Data Cleaning	Mar 10, 2019	Mar 13, 2019		Complete
Data Preprocessing	Mar 13, 2019	Mar 19, 2019		Complete
Data Augmentation	Mar 17, 2019	Mar 21, 2019		Complete
Learning Algorithms	Mar 19, 2019	Apr 15, 2019		Complete
Non- Neural Network	Mar 19, 2019	Apr 3, 2019		Complete
KNN	Mar 19, 2019	Mar 21, 2019		Complete
SVM	Mar 20, 2019	Mar 22, 2019		Complete
Naive - Byes	Mar 28, 2019	Apr 1, 2019		Complete
Decision Tree	Mar 30, 2019	Apr 3, 2019		Complete
Random Forest	Mar 30, 2019	Apr 3, 2019		Complete
Neural Network	Mar 19, 2019	Apr 15, 2019		Complete
Back Propagation NN	Mar 19, 2019	Mar 27, 2019		Complete
CNN	Apr 3, 2019	Apr 15, 2019		Complete
Evaluation 1 Report	Mar 25, 2019	Mar 28, 2019		Complete
Evaluation 2 Report	Apr 13, 2019	Apr 17, 2019		Complete
PPT	Apr 17, 2019	Apr 18, 2019		Complete
		Burndown		

State-of-the-Art Facial Expression Recognition Model

- Group of researchers from ETH Zurich (Switzerland) and KU Leuven (Belgium) point out to the fact that classifying facial expressions into different categories (sadness, anger, joy, etc.) requires capturing regional distortions of facial landmarks.
- Second-order statistics such as covariance is more suited to capture such distortions in regional facial features.
- The SOA model provides an accuracy of 87% which outperforms the baseline model by 3%.

SOA Facial Expression Recognition Model

1. Face Detection.
2. Normalized faces are fed into a deep CNN.
3. Covariance pooling is used.
4. Finally, the manifold network is employed to deeply learn the second-order statistics.
5. The covariance matrices calculated on the previous step typically reside on the riemannian manifold of SPD matrices. They are often too large. Hence their dimension needs to be reduced.



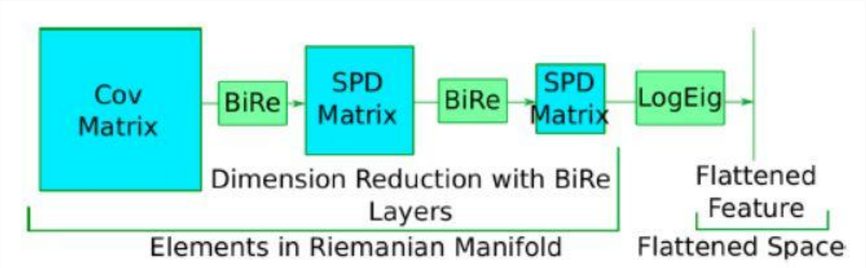


Figure 4. Illustration of SPD Manifold Network (SPDNet) with 2-BiRe layers

BiRe - Bilinear Mapping Layer and Eigenvalue Rectification Layer
→ Accomplishes the task for reducing dimensions.
LogEig - Log Eigenvalue Layer
→ Flattens the matrices, so that standard Euclidean operations can be applied.

Approach

1. Data Cleaning
    a. After importing the images, the images resized to 420 × 240 because some of the images in the dataset did not have 1280 × 960 as their size, despite the submission format.
2. Data Preprocessing
    a. The images were then converted into grayscale to remove the third dimension and to make the implementation easier.
    b. Then the images were then flattened (except for CNN) and for Neural Network we have applied PCA to reduce image's dimensions.
    c. Histogram of oriented gradients was used to extract faces from entire images.
    d. Then the dataset was divided into two parts 90% of the dataset was used for training and rest 10% was used for testing .

Approach

3. Data Augmentation
    a. We have used data augmentation to increase size of our dataset.
4. Learning Algorithms :- We have taken two types of approaches
    a. Non Neural network approach
        i. K Nearest Neighbours (with k = 5, minkowski distance with p = 2)
        ii. Support Vector Machine (linear kernel)
        iii. Naive Bayes (Gaussian with variance 10^{-9})
        iv. Decision Tree
        v. Random Forest (n = 10)
    b. Neural Network approach
        i. Back propagation Neural Network (with 15 features and 2 layers)
        ii. Convolutional Neural Network (3 convolutional layers and 2 fully connected layers with pooling layers)

Naive Bayes

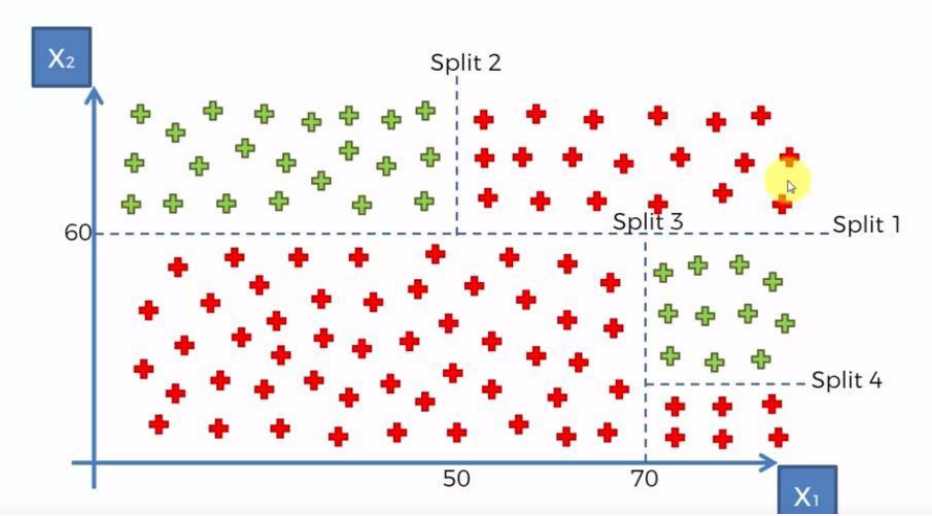
$$P\,(Emotion \mid X) = \mathrm{P(X \mid Emotion) * P(Emotion) / P(X)}$$

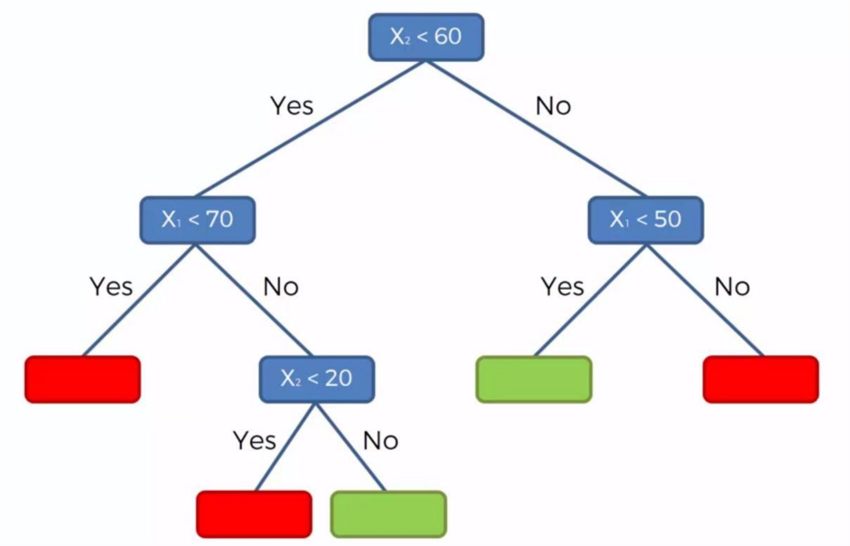
Posterior Probability *Likelihood* *Prior Probability* *Marginal Likelihood*

$$P\,(Gender \mid X) = \mathrm{P(X \mid Gender) * P(Gender) / P(X)}$$

$$P\,(Face \mid X) = \mathrm{P(X \mid Face) * P(Face) / P(X)}$$

Decision Tree





Random Forest

1. Pick random k data points from the Training set.
2. Build the decision tree associated to these k data points.
3. Choose the number Ntree of trees you want to build and repeat steps 1 and 2.
4. For a new data point, make each one of your Ntree trees predict the category to which the data points belongs, and assign the new data point to the category that wins the majority vote.

Convolutional Neural Network

Layer (type)	Output Shape	Param #
conv2d_9 (Conv2D)	(None, 48, 48, 64)	640
batch_normalization_13 (Batc	(None, 48, 48, 64)	256
activation_15 (Activation)	(None, 48, 48, 64)	0
max_pooling2d_9 (MaxPooling2	(None, 24, 24, 64)	0
dropout_13 (Dropout)	(None, 24, 24, 64)	0
conv2d_10 (Conv2D)	(None, 24, 24, 128)	204928
batch_normalization_14 (Batc	(None, 24, 24, 128)	512
activation_16 (Activation)	(None, 24, 24, 128)	0
max_pooling2d_10 (MaxPooling	(None, 12, 12, 128)	0
dropout_14 (Dropout)	(None, 12, 12, 128)	0

Convolutional Neural Network

conv2d_11 (Conv2D)	(None, 12, 12, 512)	590336
batch_normalization_15 (Batc	(None, 12, 12, 512)	2048
activation_17 (Activation)	(None, 12, 12, 512)	0
max_pooling2d_11 (MaxPooling	(None, 6, 6, 512)	0
dropout_15 (Dropout)	(None, 6, 6, 512)	0
flatten_3 (Flatten)	(None, 18432)	0
dense_7 (Dense)	(None, 256)	4718848
batch_normalization_16 (Batc	(None, 256)	1024
activation_18 (Activation)	(None, 256)	0
dropout_16 (Dropout)	(None, 256)	0

Convolutional Neural Network

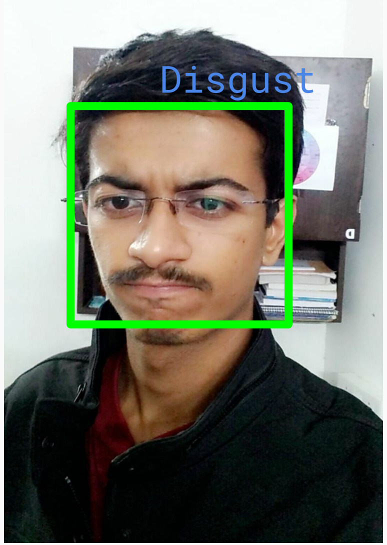
```
_________________________________________________________________
dense_8 (Dense)              (None, 512)               131584
_________________________________________________________________
batch_normalization_17 (Batc (None, 512)               2048
_________________________________________________________________
activation_19 (Activation)   (None, 512)               0
_________________________________________________________________
dropout_17 (Dropout)         (None, 512)               0
_________________________________________________________________
dense_9 (Dense)              (None, 7)                 3591
=================================================================
Total params: 5,655,815
Trainable params: 5,652,871
Non-trainable params: 2,944
```

Confusion Matrix for CNN

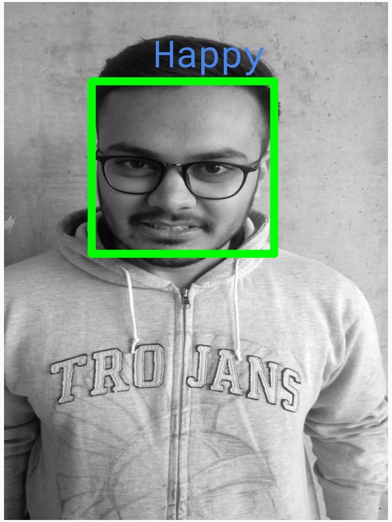
	Anger	Disgust	Fear	Happy	Neutral	Surprise	Sad
Anger	43	2	1	0	1	1	1
Disgust	1	60	2	0	2	1	2
Fear	4	3	48	2	0	3	2
Happy	0	3	0	53	1	1	1
Neutral	5	5	3	2	43	4	9
Surprise	0	0	1	2	0	63	1
Sad	1	0	2	1	4	2	44

Misclassified Images

Sad misclassified as Disgust

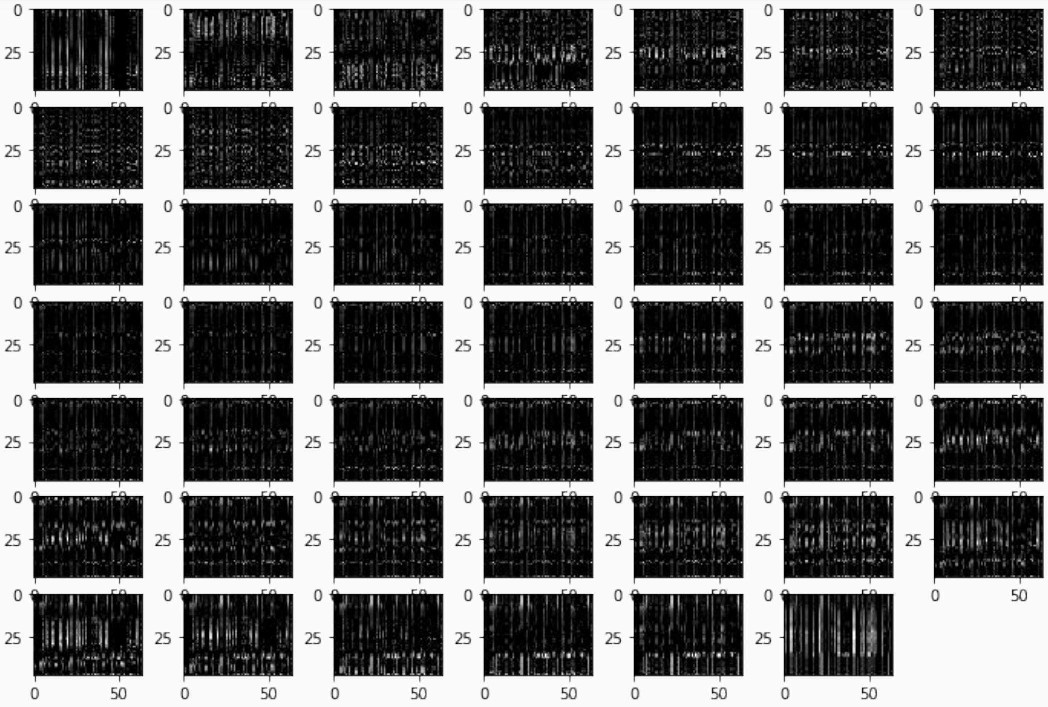


Anger misclassified as Happy



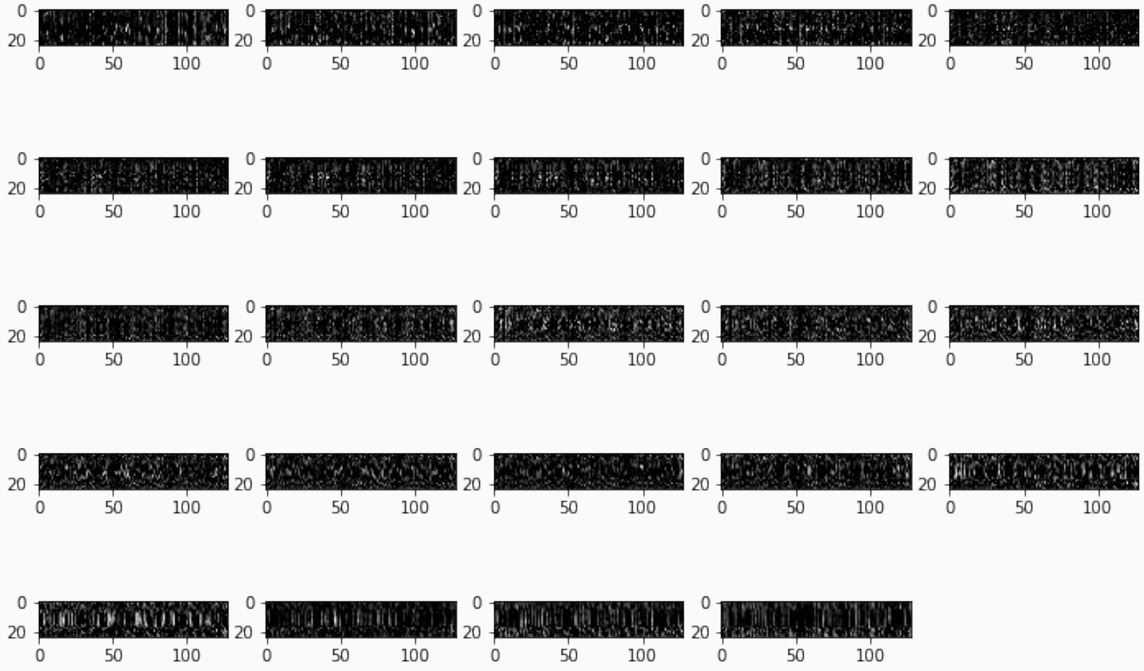
Visualization of Weights of Different Filters (Emotion Recognition)

Layer 1



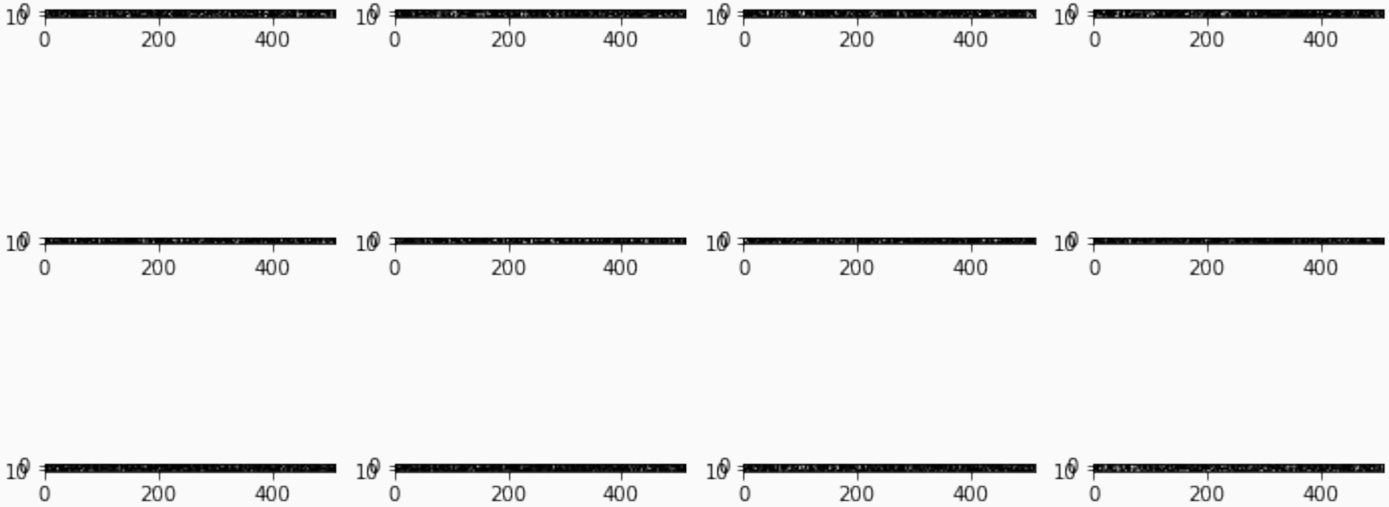
Visualization of Weights of Different Filters

Layer 2



Visualization of Weights of Different Filters

Layer 3



Final Results

Accuracy for Face Recognition

Algorithm	Test accuracy (%)
Back propagation-NN(175 epochs)	88.3
SVM	88.37%
KNN(n=5)	44.18%
Naive Bayes	46.51%
Decision Tree	37.21%
Random Forest(n = 10)	55.81%

Final Results

Accuracy for Gender Recognition

Algorithm	Test accuracy (%)
SVM	95.34%
KNN(n=5)	93.02%
Naive Bayes	93.02%
Decision Tree	88.37%
Random Forest(n=10)	93.02%

Final Results

Accuracy for Expression Recognition

Algorithm	**Test accuracy (%)**
CNN	85.20%
SVM	38.37%
KNN	11.62%
Naive Bayes	16.27%
Decision Tree	25.58%
Random Forest	26.25%

Role of each group member in the project

Name	Work Done
Nikhil	Dataset Cleaning, Data Construction and Face detection, Neural Network and **CNN**
Maharsh	Data Augmentation, Non Neural network approaches and **CNN**
Parshwa	Data Augmentation and Face detection, Neural Network and **CNN**
Krupali	Dataset Cleaning and Construction, Non Neural network approaches and **CNN**

References

1. Sharif M., Mohsin S., Hanan R., Javed M. and Raza M., ”Using nose Heuristics for Efficient face Recognition”, Sindh Univ. Res. Jour. (Sci. Ser.) Vol.43 (1-A), 63-68,(2011)
2. Maryam Murtaza, Muhammad Sharif, Mudassar Raza, Jamal Hussain Shah, “Analysis of Face Recognition under Varying Facial Expression: A Survey”, The International Arab Journal of Information Technology (IAJIT) Volume 10, No.4 , July 2013
3. https://medium.com/neurohive-computer-vision/state-of-the-art-facial-expression-recognition-model-introducing-of-covariances-9718c3cca996/
4. https://www.pyimagesearch.com/2018/06/18/face-recognition-with-opencv-python-and-deep-learning/

Thank you