Face Recognition

Prepared by: Group 07

Ashna Jain (201501008)
Dhruti Chandarana(201501015)
Charmi Chokshi (201501021)
Hena Ghonia (2015010132)
Janvi Patel (201501072)
Nishi Patel (201501076)

Subject: Computer Vision
Guided by: Dr. Mehul Raval
Date: September 19, 2018

Problem Statement

- Performing face recognition on our class dataset using three different approaches,
  - Euclidean distance based matching
  - Using SVM classifier
  - Using Convolution Neural Network

  which is robust to affine transformation (scaling, rotation, translation and illumination changes).

SIFT and HOG feature extraction

SIFT Feature Extraction:

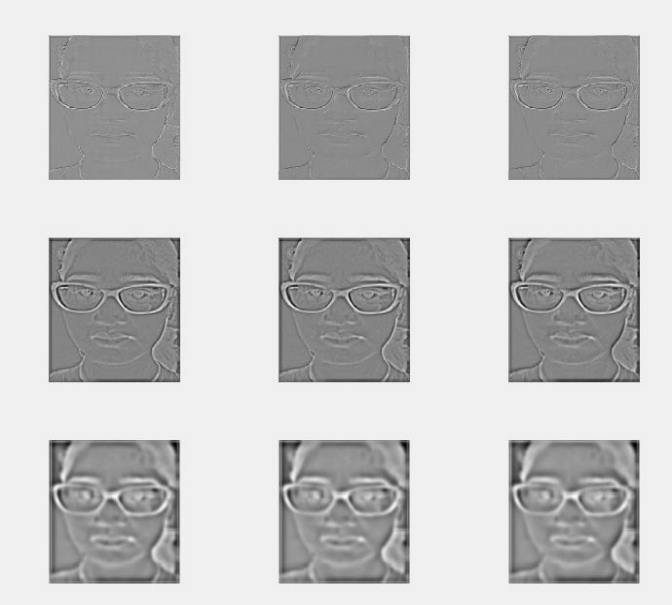
- Detecting points of interest: key points in the SIFT framework
- Scale space extrema
  - $D(x, y, \sigma) = L(x, y, k_i\sigma) - L(x, y, k_j\sigma),$
- Key Points Localization: Taylor series expansion
  - $D(\mathbf{x}) = D + \frac{\partial D^T}{\partial \mathbf{x}}\mathbf{x} + \frac{1}{2}\mathbf{x}^T\frac{\partial^2 D}{\partial \mathbf{x}^2}\mathbf{x}$
- Hessian Matrix
  - $\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$ $\boldsymbol{T_r(H)} = D_{xx} + D_{yy} = \lambda_1 + \lambda_2; \mathrm{Det}(\boldsymbol{H}) = D_{xx}D_{yy} - D_{xy}^2 = \lambda_1\lambda_2$
- Orientation Assignment

HOG Feature Extraction:

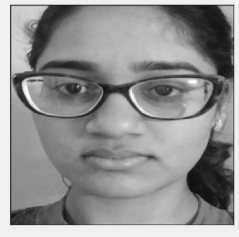
- Preprocessing
- Calculating gradient images
- Calculating Histogram of Gradients
- Calculating HOG feature vector
- Block Normalization
- Multiclass svm

Approach 1: Support Vector Machine

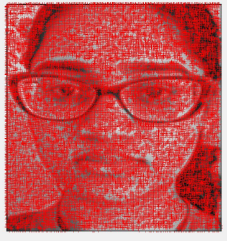
SVM classification using SIFT and HOG feature extraction



DOG Pyramid

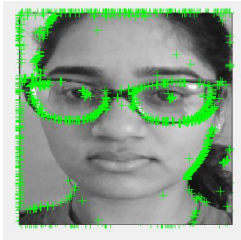


Original Image

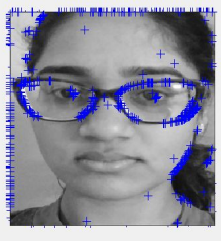


Scale-Space Extrema Detection

HOG features



keypoint localization



Eliminating edge responses

Input image

Output

Testing on validation set

```
Actual label
['10' '11' '11' '11' '12' '13' '14' '14' '14' '15' '15' '15' '19' '19'
 '20' '2' '2' '3' '4' '4' '4' '4' '5' '5' '6' '7' '7' '7' '9' '9' '9']
predicted label
['10' '11' '11' '11' '12' '13' '14' '14' '14' '15' '15' '15' '11' '7' '7'
 '2' '2' '3' '4' '4' '4' '4' '5' '5' '6' '7' '7' '7' '9' '9' '9']
Recall=  0.9032258064516129
Precision =  0.8403225806451613
Accuracy:  90.32258064516128
```

When test images are real time images as well as rotated images

```
Actual label
['11' '11' '11' '12' '13' '14' '14' '14' '15' '15' '15' '19' '19' '20'
 '22' '2' '2' '3' '3' '3' '4' '4' '7' '7']
predicted label
['7' '11' '11' '12' '13' '14' '14' '14' '15' '15' '15' '19' '7' '7' '7'
 '2' '2' '3' '7' '3' '4' '4' '11' '11']
Recall=  0.7083
Precision =  0.6749999999999999
Accuracy: 70.83
```

Approach: 2 Euclidean Distance based Matching

- Given a test image firstly, we are finding out feature vector of it.
- Then calculating distance from each image in dataset using matrix norm.
- Candidate image will be the one with minimum distance.
- For a single image,

$$x_i = \sqrt{\sum_i \sum_j (M_{ij} - V_{:j})^2}$$

$\mathrm{M} = \text{Feature matrix of training dataset}$

$\mathrm{V} = \text{Feature vector of test image}$

Results

feature Extraction

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
14.9374	11.2662	11.2399	13.2612	13.0906	11.5796	12.2061	12.3659	13.7217	14.7484	14.7609	13.8995	14.5415	14.2989	0	12.1521

Distance values for an image from the dataset

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
15.0770	10.8829	11.1403	13.1976	12.9437	11.5931	12.0871	12.1550	13.2061	13.5234	14.6621	13.9745	14.4750	14.2514	14.0849	11.9746

Distance values for an image which is not from the dataset

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
15.0908	10.1955	10.4881	12.3749	12.1297	11.2895	11.8219	11.6852	12.7287	13.5656	14.4651	13.2522	14.8515	14.3105	13.9868	11.2655

Distance values for an rotated image (not from dataset)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
15.0908	10.1955	10.4881	12.3749	12.1297	11.2895	11.8219	11.6852	13.5719	12.7287	13.5656	14.4651	13.2522	14.8515	14.3105	13.9868

Distance values for an rotated image (from dataset)

Result Analysis:

- 100% accuracy for images from the dataset
- 66% accuracy for images which are not from the dataset.
  (total 21 images, 7 not from the dataset
  Total 10 subjects
  6- images normal
  1- rotated)

Results

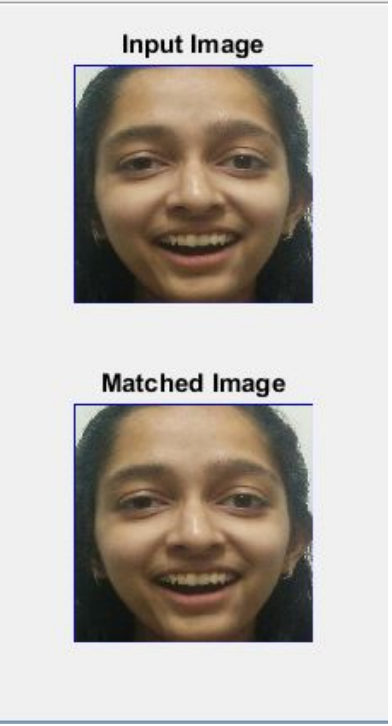


Image from Dataset

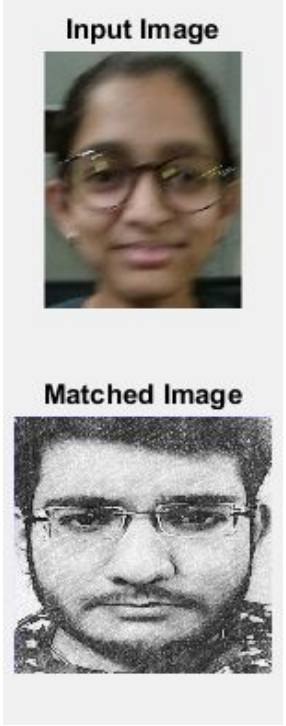
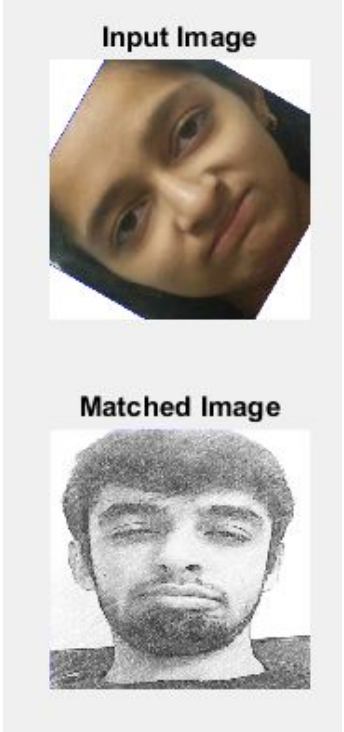
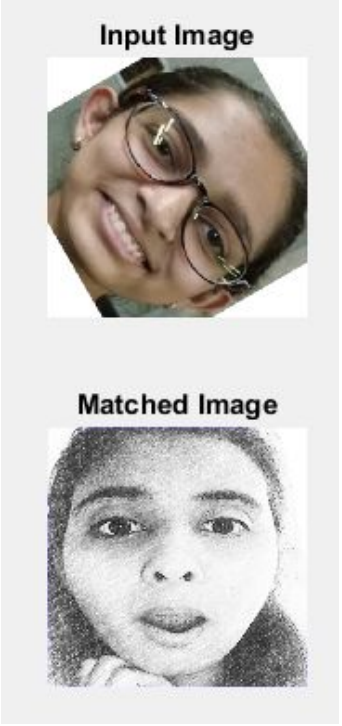


Image outside of Dataset



Rotated Image from Dataset

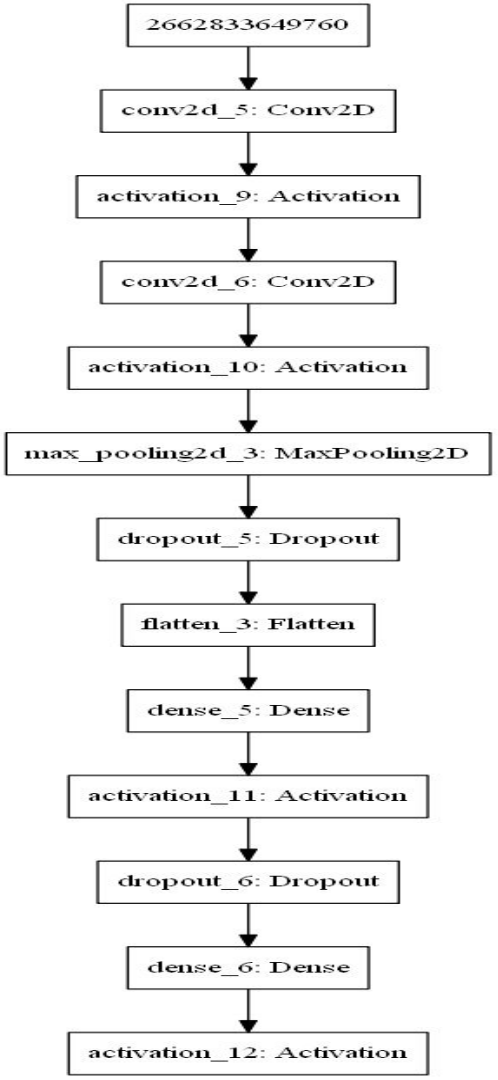


Rotated Image outside of Dataset

Convolutional Neural Network

- Preprocessing:
  - Face detection using Haar cascade classifier.
  - Files were renamed. For example 1_1.jpg, 1_2.jpg… for individual.

- Convolutional layer-
  - Number of classes- 34 and 12 images of each individual
  - Kernel size for convolutional filter: 3x3.
  - Number of convolutional filters are taken as 32.
  - Batch size: 32



- Pooling Layer
  - Max Pooling with pooling size 2x2
  - Dropout regularization is implemented where 25% of feature are deleted.
  - Adaptive moment estimation was used.

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 248, 248, 32)	320
activation_5 (Activation)	(None, 248, 248, 32)	0
conv2d_4 (Conv2D)	(None, 246, 246, 32)	9248
activation_6 (Activation)	(None, 246, 246, 32)	0
max_pooling2d_2 (MaxPooling2	(None, 123, 123, 32)	0
dropout_3 (Dropout)	(None, 123, 123, 32)	0
flatten_2 (Flatten)	(None, 484128)	0
dense_3 (Dense)	(None, 128)	61968512
activation_7 (Activation)	(None, 128)	0
dropout_4 (Dropout)	(None, 128)	0
dense_4 (Dense)	(None, 34)	4515
activation_8 (Activation)	(None, 34)	0

Total params: 61,982,595
Trainable params: 61,982,595
Non-trainable params: 0

Convolutional neural network

Training data accuracy - 97.40%
Accuracy on validation set-74%

```
346/346 [==============================] - 113s 327ms/step - loss: 1.1626 - acc: 0.7081 - val_loss: 1.2652 - val_acc: 0.7692
Epoch 6/50
346/346 [==============================] - 113s 325ms/step - loss: 0.6644 - acc: 0.8121 - val_loss: 0.9122 - val_acc: 0.8205
Epoch 7/50
346/346 [==============================] - 107s 310ms/step - loss: 0.5312 - acc: 0.8584 - val_loss: 0.6726 - val_acc: 0.7949
Epoch 8/50
346/346 [==============================] - 110s 318ms/step - loss: 0.3009 - acc: 0.9191 - val_loss: 0.7674 - val_acc: 0.8205
Epoch 9/50
346/346 [==============================] - 109s 315ms/step - loss: 0.2324 - acc: 0.9538 - val_loss: 0.5596 - val_acc: 0.7949
Epoch 10/50
346/346 [==============================] - 111s 321ms/step - loss: 0.1699 - acc: 0.9538 - val_loss: 0.7048 - val_acc: 0.7692
Epoch 11/50
346/346 [==============================] - 109s 316ms/step - loss: 0.1375 - acc: 0.9711 - val_loss: 0.5446 - val_acc: 0.7949
Epoch 12/50
346/346 [==============================] - 106s 306ms/step - loss: 0.0702 - acc: 0.9855 - val_loss: 0.5637 - val_acc: 0.8718
Epoch 13/50
346/346 [==============================] - 111s 321ms/step - loss: 0.0850 - acc: 0.9798 - val_loss: 0.4796 - val_acc: 0.8718
Epoch 14/50
346/346 [==============================] - 114s 329ms/step - loss: 0.0993 - acc: 0.9653 - val_loss: 0.6206 - val_acc: 0.7949
Epoch 15/50
346/346 [==============================] - 123s 355ms/step - loss: 0.0865 - acc: 0.9740 - val_loss: 0.6999 - val_acc: 0.7436
```

On testing data:

```
                precision    recall  f1-score   support

avg / total       0.87      0.79      0.81         39
```

Conclusions

- System designed using Euclidean distance based matching is much robust to the transformation of image (value matrix norm increases).
- SIFT and HOG give better results as compared to other algorithms, in situation where there is huge difference in the orientation of the face.
- Approach based on CNN outperforms other techniques if the data size is large. But with small data size, traditional Machine Learning algorithms are preferable.
- Model Training time: a Deep Learning algorithm based algorithm may takes weeks or months whereas, traditional Machine Learning algorithms take few seconds or hours
- Model Testing time: DL takes much lesser time as compare to ML

References

[1] David G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints”: Computer Science Department, University of British Columbia, Vancouver, B.C., Canada, lowe@cs.ubc.ca; January 5, 2004

[2] Coşkun, Musab, et al. "Face recognition based on convolutional neural network." *Modern Electrical and Energy Systems (MEES), 2017 International Conference on*. IEEE, 2017.

Thank You