
PE Report: Face & Race classification

- Suparna Ghanvatkar

Database link: <https://drive.google.com/open?id=0B15hiCPPynGsdnNLdVF4TVgzd2c>

Code: https://github.com/suparna-ghanvatkar/PE_face

Overview

Recent progress

- Experimented with face recognition in a live video based system
- Gender and Race classification

Approach

- Inbuilt OpenCV face recogniser for face recognition
- LBP histogram with SVM
- Inbuilt Face recognizer using LBP for classification
- WLD Features for race classification

Shortcoming

- Only frontal faces recognized
-

Progress - Face Recognition

Face recognition for Yale Database

- The recognizer uses inbuilt LBPFaceRecognizer and correctly recognizes all 15 subjects in the database.

Live Face recognition

- Live video based face recognition.
 - Trained on 8 people and web cam based capturing system. Recognizes the people but with a low confidence value.
-

Progress - Gender Classification

LBP histogram and SVM

- SVM on LBP histogram for detected face.
- Training set - 547 images
- Accuracy of 76.7% (13 out of 56 misclassified)

LBPHFaceRecognizer

- Inbuilt OpenCV function used.
 - 12 misclassifications of 56 images - accuracy of 78.5%
-

LBP histogram + SVM misclassifications



Misclassifications mostly for chinese
males

LBP Face Recognizer misclassifications

Most misclassifications due to SVM
approach exist in this as well

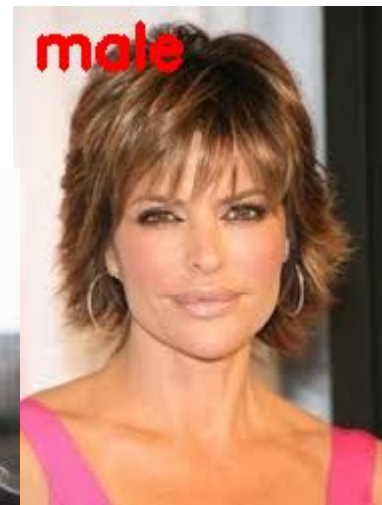
Only one female misclassified.



All images of this student
misclassified

LBP + SVM on celebrity database

4 out of 23 misclassified - accuracy of 82.6%



Progress - Race Classification

LBP histogram and SVM

- SVM on LBP histogram for detected face.
- Training set - 806 images
- Accuracy of 94.6% (9 out of 75 misclassified)

LBPHFaceRecognizer

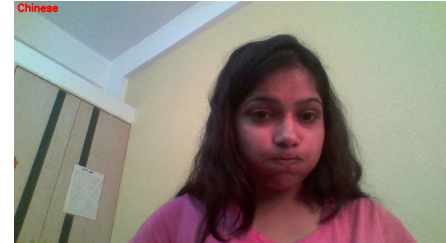
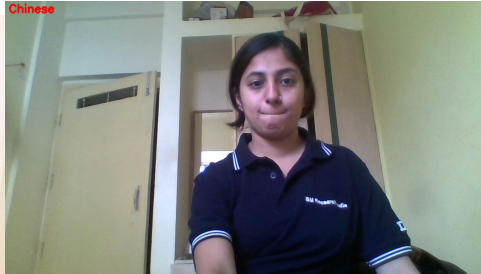
- Inbuilt OpenCV function used.
 - Accuracy of 93.33% (5 out of 75 misclassified)
 - All images from the standard datasets being identified correctly. Errors only in webcam captured new images.
-

LBP+SVM misclassifications

The faces in dataset which got misclassified:



LBPH Face Recognizer misclassifications



WLD(Weber Local Descriptor) Features

- No standard library available still - c++ code written based on the description in paper(code inspired from a github repository with partially written code)
 - Available in the folder of github repo: wld_feats_cpp
 - Calculates the WLD features for input image and outputs into feat file for the folder of images. List of names stored in the sequence of reading for identifying misclassified images. These feats are then used for classification using LinearSVR.
-

WLD feature results on previous datasets



- 1 out of 75 misclassified
- Possible overfitting!

Dataset update

1. All images cropped using haar face detection
2. Training set - 281(indian) +351(chinese) - indian images from the interview snaps and chinese photos from Caspeal dataset
3. Validation set - 112(indian)+ 105(chinese)
4. Test set - has images from IMDB for various actors, thus various get-ups, accessories, poses, expressions are present in the test set whereas the training and validation set was on more controlled dataset. This is for an attempt to generalize the results obtained
 - a. 84 chinese
 - b. 121 Indians

Folder name: cropped

WLD Features result

- Perfectly classifies all validation set results
 - LinearSVR used for labels 0-chinese and 2-indian
 - 36 misclassified out of 204 test images (accuracy of 82.3%)
 - Folder name on drive: new_results/wld_results
 - Code : race_wld.py
-

LBP Face Recognizer results

- LBPFaceReconizer used after resizing image to 180x180 with indians as label 0 and chinese as label 1
 - 136 out of 204 correctly classified on test set (66.6% accuracy)
 - So this model does not generalize well!
 - Misclassified images in folder: new_results/lbpfacerec
 - Source code: race_rec_new.py
-

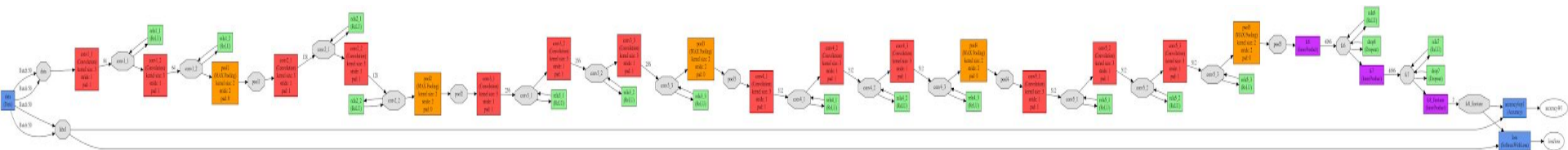
LBP Histogram with SVM

- LBPFaceReconizer used after resizing image to 180x180 with indians as label 1 and chinese as label 0
 - LBP histogram computed and classified with linear svm
 - 131 out of 204 correctly classified on test set (64.2% accuracy)
 - So this model does not generalize well too!
 - Misclassified images in folder: new_results/lbph
 - Source code: race_rec_svm.py
-

VGG Face for face recognition using Deep Learning

- Based on VGG-Very-Deep-16 CNN architecture.
 - Caffemodel i.e. the weights of the trained network provided
 - Deploy.prototxt provided which is used for inferencing
 - Problem: train_val.prototxt required for training or fine-tuning the model not provided.
-

Training Model defined by me currently



Current Status

- LMDB database used for images and mean calculated. So inputs processed and stored into train_cnn and test_cnn folders.
 - The network definition written in the train_val.prototxt now matches their network hence, now the weights can be used for fine-tuning.
 - The inner product last layer was performing classification of the people. Ankur's code performed inferencing only and used this last layer values as features for face recognition.
 - As I desire to do fine-tuning, I am performing the classification and hence have changed the number of outputs to 2 for the finetuning fc8 layer.
 - The solver.prototxt written for this network.
-

Status contd.

- The appropriate files in the github repo: VGG_FACE_train_val.prototxt and VGG_FACE_solver.prototxt
 - The network successfully gets initialized and solver scaffolding done.
 - Data prefetching start for fine-tuning the initialized network.
 - Problem: The image inputs are not matching the images on which the network has been trained. The first layer is giving an error in the transformation as the expectation for pre-trained network is different. This issue I have yet not been able to resolve though, this seems to be the issue of inputs.
-

Conclusion

1. The WLD proves to be a better feature for race classification even in not a very controlled situation but where a frontal view is available, even with some tilting of face.
 2. All the features do not seem to fare as appropriately for the low resolution or very expressive or differently accessorized faces.
-