**Applied machine learning system ELEC0132 19/20 report**

*SN: 19132626*

#### Abstract

This report has been produced as part of the AMLS (Applied Machine Learning Systems) machine learning project.[[1]](#footnote-1) It studies and analyses several different types of image classifiers and face detection models and algorithms. The gender and emotion (smiling / non-smiling) detection and classification was performed for celebA dataset of 5,000 images, using Haar Cascade face detectors and Local Binary Pattern Histograms image classifiers. The facial shape and eye color feature recognition was performed for cartoon dataset of 10,000 images use of the Face Landmark detector based on HOG. FisherFace classifier was trained to predict both gender and smiles and classical Support Vector Machines (SVM) Linear model was used for face shape and eye color detection model training and prediction.

In addition, the performance of a CNN model pre-trained by Levi and Hassner [17] was also tested for gender detection.

**Index Terms—** Face recognition, gender detection, smile detection, face shape recognition, eye color recognition

**1. Introduction**

Computer vision is among the hottest topics of today’s machine learning commercial deployments and research. Face recognition software is already actively deployed for such commercial applications as advertising, video surveillance, and user authentication. Just one of many state-of-art commercial examples is Apple’s FaceID, which uses infrared and light scans to identify a user’s face and unlock Phone X [1].

The key components of the facial recognition are person’s gender, face shapes and eye color, which have formed part of this report’s research and analysis.

In the last decade, the research has focused on the A PIE (Age Pose Illumination Expression) [2] human feature recognition. Recognition of emotions, specifically, smiling, is another topic of this paper’s analysis. It holds high promise for future applications in the areas of advertising, social media, retail and healthcare; it is an important factor of social interaction and indicator of mental health.

However, the accuracy of the models is still inconsistent. In real-life situations, environmental conditions complicate the detection and recognition of faces, and much of today’s research focuses on the face recognition “in the wild” [3]. Moreover, biases are common in face analysis to determine sex of female and darker-skinned faces [4]; recent studies found higher rates of false positives are more common for Asian and African American faces relative to images of European / Caucasian faces [5].

Until early 2000s, the methodologies for face recognition were based on supervised learning techniques, using image intensity and texture analysis with such classification algorithms as SVM (support vector machines) and AdaBoost. The article “Face recognition using eigenfaces”, published in 1991, was cited 7528 times as of early January 2020.

Recent research has focused on the use of deep learning / convolutional neural networks, which were introduced in the end of 80s/early 90s, and started being used for image, face and emotion classification in mid-2000s/2010s.

In this report, supervised learning SVM algorithm was used for face shape classification and prediction for a sub-set of Cartoon Set, comprised of 10,000 images [6]. In addition, a pre-trained deep-convolutional neural network (CNN) model was tested to determine the gender in the CelebFaces Attributes Dataset (CelebA).

This report was produced as part of the machine learning project for the UCL’s AMLS (Applied Machine Learning Systems) course .

The report is organized into 6 sections: section 2 provides an overview of literature survey on the topic of face recognition; sections 3 and 4 describes provides details of the methodology used in the project and model implementation; sections 5 and 6 present experimental results and analysis, and conclusions of the study.

**2. Literature survey**

A comprehensive study [7] of gender classification methods reviewed 120 combinations of automatic face detection, face alignment, and gender classification. It concluded that “the best classification rate was achieved with a support vector machine. A neural network and AdaBoost achieved almost as good classification rates as the support vector machine and could be used in applications where classification speed is considered more important than the maximum classification accuracy.”

The use of SVM and AdaBoost classifiers was applied directly to image intensities by [8], [9]. [10] experimented on the popular Labeled Faces in the Wild (LFW) [11] benchmark, using a combination of LBP features with an AdaBoost classiﬁer.

FERET database [12] has now been established as a performance benchmark for gender classification. Recently, [13] used the Webers Local texture Descriptor [14] for gender recognition, demonstrating near perfect performance on [12]. Similarly, [15] obtained near-perfect result on the FERET benchmark using intensity, shape and texture features.

CNN approach, initially used for optical character recognition by Y.LeCun et al in 1989 [16], has now been extensively used for image classification as the availability of data and compute power improved. The CNN model described in this report and tested for gender detection task was described by Levi and Hassner in [16]. This paper also provided an overview of other neural network approaches, noting that most approaches to detecting or classifying gender attributes have used differences in facial feature dimensions [17] or “tailored” face descriptors [18]. It concluded that early methods for gender classification [19] had used a neural network trained on a small set of near-frontal face images. This focused has changed to the detection of facial recognitions in more challenging environments, for example, in [21].

For the emotion detection and facial expression recognition, this paper set out to apply Fisherfaces and Local Binary Pattern Histograms (LBP), based on the high accuracy reported for these in the surveyed literature.

A high-level overview of research to date concluded that most common approaches included static image texture analysis, feature landmark-based expression classifiers, 3D face modelling [21], and dynamic analysis of video sequences.

The classical machine learning approaches to face recognition have used five popular image classifiers, including Eigenfaces (1991, which uses Principal Component Analysis (PCA)), Fisherfaces (1997, which uses Linear Discriminant Analysis (LDA), Local Binary Pattern Histograms (LBP), 1996, which uses texture based descriptors, Scale Invariant Feature Transform (SIFT) (1999) and Speed Up Robust Features (SURF) (2006). LBP in particular has been widely used due to its computational simplicity and good performance. The concept of LBP was made popular following the paper published by Ojala et al [21].

Whitehill et al [22] focussed on smile detection and optimising performance in real-time real-world imaging conditions. They tested performance of 3 image representations -- Gabor Energy Filters, Box Filters (e.g. Haar), and Edge Orientation Histogram – on SVM and GentleBoost algorithms, using the GENKI database of 36,000 images of real-life faces.

More recently, emotion detection use deep learning approaches, and focus on improving the accuracy of emotion recognition in difficult light conditions [23], in real-time situations, and on small datasets, using transfer learning [24].

The face detector used in this report uses the classic Histogram of Oriented Gradients (HOG) feature combined with a linear classifier, an image pyramid, and sliding window detection scheme. Shape detection research has been driven by the need to recognize shapes for face recognition, and especially by active research in the area of autonomous vehicle driving.

The pose estimator was created by using dlib's implementation of the Kazemi and Sullivan paper [25], and was trained on the iBUG 300-W face landmark dataset [26]

…

Finally, this paper has investigated eye color classification using hue, saturation, value (HSV) using a classical SVM algorithm. To aid with the eye detection, the Hough Transform, a global transform to extract features, can be used to detect circles and ellipses in the image. It is also possible to use Haar Cascades for face and eye identification.

A potential eye color detection approach is based on the approach developed by Tristan Hume [27] based on the research by Timm and Barth [28], which uses image gradients and dot products to create a function that is at a maximum at the center of the image’s most prominent circle. This function has been developed specifically for difficult scenarios, e.g. low resolution and low contrast. Eye state detection focusing on iris was also investigated in [31].

**3. Description of models**

Several image classifiers, face detection models and classification algorithms were used for the project’s 4 tasks:

* Haar Cascade face detectors, Face Landmark detector based on HOG developed by [26], and Local Binary Pattern Histograms were used to detect and recognize faces;
* A Fisherface classifier was trained to predict both gender and smiles in A1 and A2 task; a CNN model pre-trained by [17] was also tested for gender detection A1 task. Classical Support Vector Machines (SVM) Linear model was used for face shape detection/B1 task and eye colour detection/B2 task.

The choice of classifiers and models was based on the model effectiveness evaluation in the surveyed literature. The objective behind using several different models for this project was to understand the workings, and to test the implementation of each model.

**3.1. Task A1: Gender detection: male or female**

The gender detection task used the CelebA dataset of 5,000 images. Two models were compared in this task:

Model 1 used Haar cascade image representations to detect faces; two face recognition filters -- Fisherface and LBP face recognizers – were tested. The Fisherface model was then trained on 80% of data and applied to 20% of data to predict gender.

Model 2 was used to predict gender representations using a pre-trained deep-convolutional neural network (CNN). The rational for choosing this model was the interest in testing how a pre-trained CNN model would perform when applied to a new dataset, the availability of the model itself in open access, as well as is its compact size (44.5MB) and computational efficiency.

The convolutional neural network model was trained by [17]; using 3 convolutional layers:

Convolutional layer; 96 nodes, kernel size 7

Convolutional layer; 256 nodes, kernel size 5

Convolutional layer; 384 nodes, kernel size 3

The model has 2 fully connected layers, each with 512 nodes, and a final output layer of softmax type.

The authors of [17] noted that they tested its model on the Adience benchmark for gender classification of unfiltered face images and “outperformed existing state of the art by substantial margins.” However, the authors also noted that there was further room for improvement, and the trained model was provided for public access.

Fig. 1 Gender Detection Model Flow

**3.2. Task A2: Emotion detection: smiling or not smiling**

CelebA dataset of 5,000 images was used for emotion detection task. Similar to the gender prediction task, prior to training the model, the images were processed using Box Filter image representations, namely Haar features that use the Viola-Jones algorithm, such as Haar face, eyes and smile cascades to detect a face and a smile. OpenCV was used for Haar cascades and we tested face recognition using Fisherface and LBP face recognizers. This was based on the conclusion from the Whitehill paper [22], that the use of LBP was shown to improve the performance, compared with the use of Box Filters alone.

To detect smiles, we trained the model on 4,000 random images to recognize smiles and predicted the labels on the remaining 1,000 images.

**3.3. Task B1: Face Shape recognition: 5 types of face shapes**

Cartoon set of 10,000 images of avatars was used for face shape recognition. SVM linear model was used for classifying the images. The model was trained on 7,500 images (75% of the total set) and predicted the labels on the remaining 2,500 images (25% of the total set).

Prior to training the model, Dlib frontal face detector was used to find frontal faces using 68 landmarks.

###### 3.4. Task B2: Eye color recognition: 5 types of eye colors

Hello world!

Fig. 1 A nice view of Roberts Building

**4. Implementation**

The implementation of the models was structured with the following flow:

1. Import of Python libraries (os for image and labels file paths, cv2, csv for labels files read/write, dlib for image processing and , NumPy for image array manipulation, Sklearn for SVM, train-test-split, accuracy score modules, Keras for image processing operations.
2. Access and manipulate images:
   1. All images were converted to Grayscale using cv2 cvtColor function. This is because Haar cascades, which were used as facial detectors, use the Viola-Jones algorithm, which works with grayscale images.
   2. CLAHE (Contrast Limited Adaptive Histogram Equalization) was applied to improve image contrast.
3. Face detection and recognition using Haar cascades, BLP classifiers and face Landmark based on HOG.
4. The dataset was split into a training set (80%) and prediction set (20%) based on a random selection of data in the full dataset.
5. The predictions were made using a trained FisherFace classifier for gender / emotion detection; and using SVM for face shape and eye colour prediction.
6. Model accuracy was calculated for each test set.

**4.1. Task A1: Gender detection: male or female**

The Python implementation of the model used the following libraries: OS - for file manipulation, cv2 - for image operations and running DNN model, and CSV for labels file manipulation

* + 1. *Implementation of the gender model – CNN*

The open access Caffe model for gender classification was provided by Tal Hassner on his web site [30]. The method was implemented using the Caffee open-source framework.

The Python code for the CNN gender\_net model was modified from the code provided in N.Singh Chauhan post [30] for TowardsDataScience site.

The images were processed using a selection of 5 Haar cascade frontalface classifiers in OpenCV library, including frontal face and profile classifiers. OpenCV dnn package was used to deploy the model. The dnn package uses a class called Net, used to populate a neural network. Data augmentation is used by image crop to 227x227 pixels, and the training is performed using stochastic gradient descent with image batch size of 50 images.

The model uses 3 convolutional and 3 fully connected layers. According to [30], “the output of the last fully connected layer is fed to a soft-max layer that assigns a probability for each class. The prediction itself is made by taking the class with the maximal probability for the given test image.”

* + 1. *Implementation of the gender model – Fisherfaces classifier*

The Python implementation of the model used the following libraries: OS - for file manipulation, cv2 - for image operations, and CSV for labels file manipulation, dlib - for image processing; NumPy - for working with image arrays; and random - for randomly splitting datasets into training and test sets. The Python code used snippets of the code published by Paul van Gent [32].

A model was then trained on 4,000 images using existing labels, and “smiling / non-smiling” label predicted on 1,000 images. A random selection of images in the full dataset was split into 80% for training and 20% for testing.

**4.1. Task A2: Emotion detection: smiling or not smiling**

The Python implementation of the model used the following libraries: OS - for file manipulation, cv2 - for image operations, and CSV for labels file manipulation, dlib - for image processing, NumPy - for working with image arrays, and random - for randomly splitting datasets into training and test sets.

Two different face recognizers – FisherFace and LBP – were used to recognize faces.

LBP computes a local representation of texture, comparing each pixel with its nighbouring pixels and calculating and storing the LBP value in output 2D array.

A model was then trained on 4,000 images using existing labels, and “smiling” label predicted on 1,000 images. A random selection of images in the full dataset was split into 80% for training and 20% for testing.

The code for the detection used modified code from the implementations in the Python Smile Detection Using OpenCV [23] and How to Build a Smile Detector [ [24] articles.

Code for LBP – Model training was adapted from Paul van Gent’s article [ [25]]

**4.1. Task B1: Face shape recognition: 5 types of face shapes**

For the face shape recognition, we used the SVM model to train the model and predict face classification.

Features are extracted using the Landmark model, which is based on the HOG (histograms of oriented gradients) feature extractor combined with a linear classifier, an image pyramid, and sliding window detection scheme. The pose estimator was created by using dlib's implementation of the paper [26].

The following were extracted for each image and returned:

* landmark\_features: an array containing 68 landmark points for each image in which a face was detected.
* shape\_labels: an array containing 5 face shape labels (0,1,2,3,4) for each image in which a face was detected.

The code implementation was based on the SVM materials provided in Lab2 of the UCL’s ELEC0136 course.

“An image is divided in a grid fashion into cells, and for the pixels within each cell, a histogram of gradient directions is compiled. To improve invariance to highlights and shadows in an image, cells are block normalized, meaning an intensity value is calculated for a larger region of an image called a block and used to contrast normalize all cell-level histograms within each block. The HOG feature vector for the image is the concatenation of these cell-level histograms.

**4.1. Task B2: Eye color recognition: 5 types of eye colors**

Hello world!

**5. Experimental Results and Analysis**

This section describes and discusses your results. Additionally, this section should include accuracy prediction scores on a separate test dataset, provided by the module organizers, but not used during your training and validation process.

We recommend you use a table to list the tasks, models and results before analysis.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Task | Model | Face recognition Acc | Train Acc | Val Acc | Test Acc |
| A1 | Caffe model | n/a |  |  | 52.62% |
| A1 | Fisherface | 94.7% |  |  | 83.47% |
| A2 | Fisherface | 95.3% |  |  | 70% |
| B1 | SVM | 82% |  |  | 69% |
| B2 | SVM | n/a |  |  | TBD |

**5.1. Gender Detection Analysis**

The first run of the gender detection pre-trained CNN model yielded very poor accuracy of 52.62% - barely over a random guess, and well below the results achieved in the source paper (85.9% +/-1.4 using single image crop).

A selection of misclassified images is provided below.

Fig. 1 Gender misclassifications

**5.1. Emotion Detection Analysis**

The accuracy of the emotion recognition model (50%) was calculated based on an average of 10 runs. Of these, 4 runs were performed using FisherFace recognizer, with an average accuracy of 42.68%; and 6 runs using LBP recognizer with an average accuracy of 55.63%.

The accuracy of the predictions was affected by the accuracy of face recognition – this was 95.3% on average.

These results were lower than results reported in Whitehill et al [22], where a combination of image representation – classifier reported where 96.3% +\_0.27 was reported for smile detection.

**Task B1: Face shape recognition:**

**6. Conclusion**

This last section summarizes the findings and suggests directions for future improvements.

**12. References**

List and number all bibliographical references at the end of the paper. The references can be numbered in alphabetic order or in order of appearance in the document. When referring to them in the text, type the corresponding reference number in square brackets as shown at the end of this sentence [2]. An additional final page (the fifth page, in most cases) is allowed, but must contain only references to the prior literature.

[1] A.B. Smith, C.D. Jones, and E.F. Roberts, “Article Title,” *Journal*, Publisher, Location, pp. 1-10, Date.

1. The code for the project is provided in the following GitHub depository: <https://github.com/sgrantcs/-AMLSassignment19_20.git> [↑](#footnote-ref-1)