Proceedings of the Second International Conference on Inventive Systems and Control (ICISC 2018)
IEEE Xplore Compliant - Part Number:CFP18J06-ART, ISBN:978-1-5386-0807-4; DVD Part Number:CFP18J06DVD, ISBN:978-1-5386-0806-7

Implementation of Machine Learning for Gender Detection using CNN on Raspberry Pi Platform

Mitulgiri H. Gauswami
Communication System Engineering
Electronics & Communication Engineering
Shantilal Shah Engineering College, Bhavnagar, India
Gujarat Technological University, Ahmedabad, India
mitul.giri@gmail.com

Kiran R. Trivedi
Associate Professor
Electronics & Communication Engineering
Shantilal Shah Engineering College, Bhavnagar, India
Gujarat Technological University, Ahmedabad, India
krtrivedi@gmail.com

***Abstract*—** Gender Detection has numerous application in the field of authentication, security and surveillance systems, social platforms and social media. The proposed system describes gender detection based on Computer Vision and Machine Learning Approach using Convolutional Neural Network (CNN) which is used to extract various facial feature. First, the facial-extraction is investigated and best features are introduced which would be useful for training and testing the dataset. This learning representation is taken through the use of convolution neural network. Which reveals that the proposed system is tested across various challenging levels of face datasets and gives excellent performance efficiency of the system with gender detection rate for each of the database. This whole system is introduced by the simple and easy hardware implementation on Raspberry Pi programmed using Python.

Keywords—*Machine Learning, Gender Detection, Google Cloud Vision API, Raspberry Pi, Convolutional Neural Networks(CNN), Artificial Intelligence, Linux Platform, Embedded System.*

I. INTRODUCTION

The human eye is the vital part of the human visual system which provides a three dimensional, moving image, normally colored in daylight. it also extracts some features from different images that the decision is to be taken for what the image is all about. Nowadays, the computer is being trained in such a way that it can predict some specific result by taking images as input which works like the human visual system, hence it refers to as computer vision technology. Computer vision technology [19] can be defined as the science and technology of the machines which are able to collect and analyze images or videos with the aim of extracting image features from the processed visual data and concerned with the theory behind artificial intelligence - system. This system seeks to apply its theories and models for implementation of computer vision. In recent year the cameras are becoming smart as they possess standard computer hardware and required features like mobile devices. Computer vision is useful tool to move toward wide range of applications with the aims of different algorithms and frameworks such as social media platforms, industrial robots, event detection, image analysis (e.g. face recognition, medical image analysis), information management systems as well as input for human-computer interaction devices.

This paper aims to review the Google's cloud vision technology which is used to compute the contents of the images through powerful machine learning processes. This solution permits users to extract some relevant information from the visual data containing image labeling, face and landmarks detection, optical character recognition(OCR). By using the REST API, it is then easy to interact with Google's cloud vision platform, called Google Cloud Vision API [2] [3]. In this paper we are going to exploit embedded system and software resources in order to fulfill the gap of gender detection for Google Cloud Vision technology. Here we elaborate the design and real-time implementation of the hardware as well as software solution we made by using low cost Raspberry Pi 3 model B+ board with Pi Camera module [17], which itself minicomputer like credit card size and like a portable device. The following embedded system includes a specialized software tool for image processing (e.g. python) [2]. Afterward best facial features are to be introduced for training and testing the dataset in order to achieve improved gender detection performance rate for each of the dataset. A recently introduced Adience benchmark face database [7] is to be taken for training and testing purpose. We propose that by learning representation through the use of convolutional neural network(CNN), there is senseful increase in efficiency or say performance can be obtained on this work. We show that despite the very challenging nature of the images in the Adience dataset, the proposed method outperforms existing innovation by substantial margins.

II. RELATED WORKS

Different works are done on facial gender detection which introduce unique results with their performance rate for different database. Those methods rely on the following causes: what are the basis for the face features extractions? How will be done the analysis of extracted features and results? What type of sample database have been taken. Then after the gender detection process is carried out. [4]

H. D. Vankayalapati [4] has contributed his work based on Support Vector Machine(SVM) algorithm for feature classification using MATLAB. Facial edge has carried out using Laplace of Gaussian filter to determine the landmarks positions. GTAV face database is used for the verification of input data. The limitation of this work is, the classification may differ with the human race [4]. Hence to eliminate this limitation of race and ethnicity Elham Arianasab [5] presented his work using Neural Network-based classification algorithm for gender diagnosis and reliability is mainly based on pixel value and geometric facial features. For the robustness of this system, training and testing on whole dataset is presented to classify them into male and female using Neural Network.

The classification accuracy was also affected by resizing the face images before and after alignment [6]. Erno Makinen and Roope Raisamo [6] have proposed four fundamental different gender detection method like SVM [4], LBP, Adaboost and Neural Network with their classification rate and sensitivity analysis for classifiers by varying rotation, scale, and translation of the face images by using IMM face database as well as FERET database. Gil Levi [7] presented the Convolutional Neural Network (CNN) for different face positions, pixel resolutions and size which shows noticeable increase in performance of gender classification rate. The Adience face dataset has been used for training and testing the particular dataset.

Finally, for the real-time application purpose most preferable and reliable board for gender detection system, Raspberry Pi 3 Model B+ board and camera module has been used by Davide Mulfari [2] for making an assistive technology system by using Google Cloud Vision platform's REST API to process images as well as facial features extraction in form of JSON response [2][3] which is then used as a database for learning purpose.

Similarly, we'll conduct the same implementation using CNN as well as Raspberry Pi board which itself a mini-computer for real time application to close the gap of Google Cloud Vision technology.

III. PROPOSED WORK

Embedded System

This paper demonstrates a real-time application for the gender detection by using Raspberry Pi 3 board with camera module. The camera module interfaced with Raspberry Pi board extracts information related to input images by using computer vision library. [2]

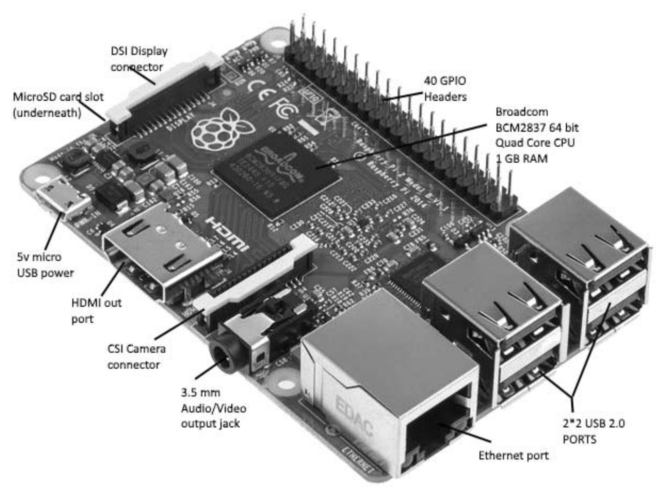

Fig. 1. **Raspberry Pi 3 board**.

This section presents details on hardware and software components. Both the hardware modules have shown in Fig. 1 and Fig. 2 respectively. Raspberry Pi 3 module is an inexpensive Linux based mini-computer. Which has 40 GPIO Pins to control hardware such as LEDs, motors and relays which are output components. This module has following hardware specifications:

- System on-chip: BCM2837
- CPU: 1.2 GHz quad-core ARM Cortex A53
- GPU: Broadcom Video-core IV @400 MHz

- Memory: 1 GB LPDDR2-900 SDRAM
- USB Port: 4
- Network: 10/100 Mbps Ethernet,
- 802.11n Wireless LAN and Bluetooth 4.0

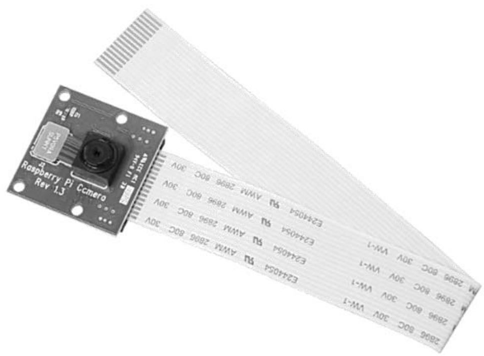

Fig. 2. **Raspberry Pi Camera Module**

Fig. 2 shows the raspberry pi camera module which has pixel resolutions of 2592 x 1944 pixel, connects by way of 15 pin Ribbon Cable to dedicated 15 pin Camera Serial Interface(CSI), specially design for camera module. This CSI bus is capable of extremely high data rate. Raspberry pi module weight is about 3g, dimension at 25mm x 20mm x 9mm, hence board itself is tiny and perfect about size and weight, which is very important for mobile and other applications.

From the software point of view, our embedded system loaded with the Raspbian Linux distribution which is most popular operating system consider for our hardware module. A Python software has been used for complete the following tasks: i) capturing an image for face detections; ii) Extracting a facial feature and analysis of face database; iii) training and testing of particular face dataset; iv) process of Convolutional Neural Network (CNN); v) classification of dataset which shows the results either male or female. Fig. 3 shows the block diagram for these processes which has been implemented in Python.

Network Architecture

Figure 4 shows the Network Architecture of this system based on work in [7, 8]. To prevent the overfitting the data, this network intended to design for particular system. The whole system is explained below:

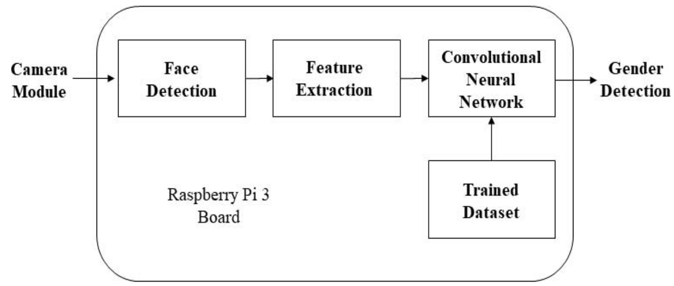

Fig. 3. **Block Diagram**

An RGB image is captured through Raspberry Pi camera module which is first scaled to 3 x 256 x 256 and then cropped to 3 x 227 x 227. This cropping is further detailed in the next session. Three convolutional layers and three fully connected layers are described as follows.

1. 96 filters of size 3 x 7 x 7 pixels are applied to the input image in the convolutional layer - 1 with 4 strides and zero padding, resulting output of size 96 x 96 x 56, which is followed by a ReLU, max-pooling to reduce the size to 96 x 28 x 28, and a Local Response Normalization (LRN).
2. The output of first is applied to convolutional layer - 2 with 256 filters of size 96 x 5 x 5 convolved with 1 stride and 2 padding, resulting output of size of 256 x 28 x 28. Which is further followed by ReLU, max-pool and LRN, reducing the output size to 256 x 14 x 14.
3. This second output is applied to convolutional layer - 3 with again 256 filters of size 256 x 3 x 3 are convolved with 1 stride and 1 padding, resulting in an output of 256 x 7 x 7 sizes.

The fully connected layers are described as:

4. The first fully connected layers which receives the output of third convolutional layer, has 512 neurons followed by a ReLU and dropout layer.
5. The second fully connected layer of 512 neurons fully connected to the 1 x 512 output of first fully connected layer followed by a ReLU and dropout layer.
6. The final fully connected layer with 2 or 8 neurons fully connected to the 1 x 512 output of second fully connected layer which maps to final classes of gender detection.

Fig. 4. **Network Architecture for CNN [8]**

Finally, the dropout layer of final fully connected layer is fed to a soft-max layer that gives a probability for each class with loss.

Training and Testing of Dataset

The weights are initialized with random values from a 0 mean Gaussian with standard deviation of 0.01 in all the layers. Here we have not used the pretrained model for initializing the network. Our network is trained from the scratch, without using any information exterior of the images and the labels available by the benchmark. This, further, should be differentiate with CNN implementations which is used for face recognition. There are hundreds of thousands of images are used for training [7, 8].

A sparse and binary vector corresponding to the ground truth classes representation is used for the target value of trained dataset. Here the target or label vector have two classes for gender detection (e.g. Male or Female) for each trained image which contains one for the index of the ground truth and zero otherwise. This paper aims to limit the risk of overfitting. Hence for that point of view two methods are used. Which are as follows.

1. **Dropout layers:** The two dropout layers are introduced with ratio of 0.5 to set the neuron's output value to 0.
2. **Data-Augmentation:** Taking a random crop of 256 x 256 input image into 227 x 227 pixels image and randomly mirror it in each forward-backward training pass.

The stochastic gradient decent optimizer is used for the dataset to be trained by itself with 50 images batch size. After 10K iteration the learning rate is reduced to e^{-4} from e^{-3} (initial learning rate).

In order to produce the gender prediction for different – faces, two methods of using network are introduced as follows.

1. **Center Crop:** Images are cropped to 227 x 227 around the face center which are used to feed the network.
2. **Over-sampling:** five 227 x 227-pixel crop regions, four from the corners of the 256 x 256-pixel face image with additional center crop of the face are extracted. From all these variations the average prediction value is considered as a final prediction.

There is a noticeable impact on the quality of the result for the small misalignments in the Adience face images caused by challenges like occlusions, motion blur etc. therefore to compensate for this small misalignment, the oversampling is used to improve the alignment quality.

Technical Details

The technical details related to our network architecture and trained model are elaborate as below:

Local Response Normalization (LRN):

The Local Response Normalization layers [9] are used here after first two pooling layers which is used to help the generalization of CNNs. The main reason behind LRN is for introduction of lateral inhibition between the various filters for the given convolution by making them "compete" for large activations over particular segment of their input. This affects to prevent repeated recording of the same information. Here, if $a^i_{x,y}$ is the activation of a neuron by applying kernel i at position (x, y), then it's LRN activation $b^i_{x,y}$ is as follows:

$$b^i_{x,y} = a^i_{x,y} / \left(k + \alpha \sum_{j=max(0,i-n/2)}^{min(N-1,i+n/2)} (a^j_{x,y})^2 \right)^{\beta}$$

Here, k, n, α, and β are the hyper-parameters. n is the number of "adjacent" kernel filters. N is total numbers of kernels in that given layer.

Softmax function:

Softmax function is used after the final fully connected layer for which is used to compute the loss term and also used to optimize during training and the class probabilities during a classification. This function is also known as multinomial logistic regression. Suppose we have $z_{i,}$ is the score assign to class i after the final fully connected layer, then the softmax function [9] is defined as follows:

$$f_j(z) = \frac{e^{z_j}}{\sum_k e^{z_k}}$$

Because we want to maximize the log likelihood of the correct class. Now here we have to minimize the negative log likelihood.

$$L_i = -log(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}})$$

Because the softmax function is used to takes real-valued score being output from f and normalizes them by their exponentiated sum, it suggests that the sum of all softmax scores is 1. It should be considered that the softmax loss is actually a particular form of a cross-entropy between an actual distribution p and an approximate distribution q is as follows:

$$H(p,q) = -\sum_x p(x) logq(x)$$

From this function we can see that softmax classifier is used to minimize the cross-entropy which would look like one predicted for actual class and zero predicted for everything else.

Stochastic Gradient Descent:

After finding the loss, we need to require how to minimize it in order to train an accurate classifier. For this experiment we are going to optimize this by using Stochastic Gradient Descent function [9]. For this function first, we need to know about gradient which is basically derivative of loss function with respect to all the variables/ weights. Then we will have the direction along which we can move toward our minimum loss most quickly by following the negative of the gradient [8, 9]. Each of the time we will compute the gradient we take a small step in the opposite direction an we re-evaluate the loss, re-compute the gradient, and repeat. Hence, we will decrease our loss function by repeating this process iteratively therefore better its classification work. Mathematically we can describe this as follows:

$$w = w - \eta \ \delta_w L$$

where η is the learning rate or also called the step size and $\delta_w L$ is the gradient of the loss term with respect to the weight vector w.

IV. DATASET

The dataset used for testing and training for this implementation is Adience dataset which is recently introduced by Open University of Israel (OUI) has approximately 26K face images with 2,284 unique subjects and are collected from Flickr [7, 8]. Various challenging levels like occlusion, lighting, and blur are subjected with all these images. Here I have used mostly front face images near about 20K images. This dataset is also used for gender as well as age prediction system hence Table-1 shows the distributions of images in each gender and age respectively. These all images are of 768 x 768-pixel size which are being resized to 256 x 256. Fig. 5 shows some example of images of both males and females.

	0-2	4-6	8-13	15-20	25-32	38-43	48-53	60+	Total
Male	745	928	934	734	2308	1294	392	442	8192
Female	682	1234	1360	919	2589	1056	433	427	9411
Both	1427	2162	2294	1653	4897	2350	825	869	19487

Table-1 **Adience Image Dataset distribution**

Fig. 5. **Various Face images from Adience dataset [7]**

V. EXPERIMENTS

The proposed work is implemented on Raspberry Pi board for the real-time application which can be shown in Fig. 6 as our system prototype. It shows the system arrangement for perfect face and gender detection.

VI. CONCLUSION

Google has developed an extraordinary computer vision technology in the last year which has introduced a specialized REST API also called Google Cloud Vision

API. By using this, Developer can remotely access in easy way to process the content of face images in order -

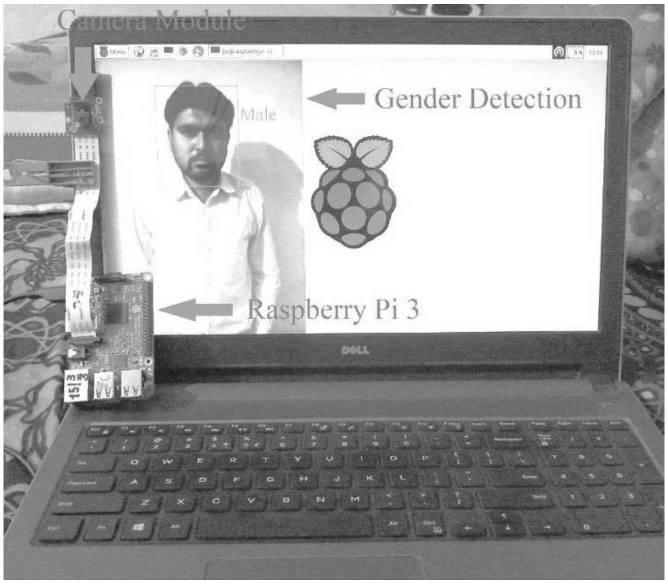

Fig. 6 **Prototype for Gender Detection System**

to extract some information from visual data with face and landmark data to explore their work. In this paper we have discussed the real-time application of gender detection to close the gap of Google Cloud Vision technology which has given the facial features only. But by using these features we have elaborated our work in the direction of CNN for implementation of gender detection to accurately predict the class of given data (either male or female) on very cheap and credit card sized processor Raspberry Pi board equipped with camera module. We believe that this project is a very innovative for the computer vision technology. Here we have presented the design of prototype for gender detection system which will be very helpful in the field of security or say authentication system to identify the person's gender. For these reasons, in future works, we plan to explore our system to identify the person from their movement as well as their facial properties. We also plan to modify our system prototype in such a way that to execute on wearable devices (e.g. smart glasses) equipped with on-board camera with remotely processed the captured visual data over the cloud.

REFERENCES

[1] Vladimir Pavlov, Vladimir Khryashchev, Evgeny Pavlov, Lev Shmaglit, "Application for Video analysis based on Machine Learning and Computer Vision Algorithms", *Yaroslavl State University, Yaroslavl, Russia*.

[2] Davide Mulfari, Antonio Celesti, Maria Fazio, Massimo Villari and Antonio Puliafito, "Using Google Cloud Vision in Assistive Technology Scenarios," *978-1-5090-0679-3/16/$31.00 ©2016 IEEE.*

[3] Hossein Hosseini, Baicen Xiao, Radha Poovendran, "Deceiving Google's Cloud Video Intelligence API Built for Summarizing Videos," *2017 IEEE Conference on Computer Vision and Pattern Recognition Works.*

[4] H.D Vankayalapati, L N P Boggavarapu, R S Vaddi, K R Anne, "Extraction of facial features for the real-time human gender classification," Department *of Computer Science and Engineering V R Siddhartha Engineering College Vijayawada, India,* 978-1-4244-7926-9/11/$26.00 ©2011 IEEE.

[5] Elham Arianasab, Mohsen Maadani, Abolfazl Gandomi, "A Neural Network Based Gender Detection Algorithm on Full-face Photograph," 2015 2nd *Interenation Conference on Knowledge Based Engineering and Innovation November 5-6, 2015, 978-1-4673-6506-2/15-IEEE.*

[6] Erno Makinen and Roope Raisamo, "Evaluation of Gender Classification Methods with Automatically Detected and Aligned Faces," *IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 3, pp. 541-547, March 2008.*

[7] Gil Levi and Tal Hassner, "Age and Gender Classification using Convolutional Neural Networks," *IEEE Conference on Computer Vision and Patern Recognitions, boston, 2015.*

[8] Ari Ekmekji, "Convolutional Neural Network for Age and Gender Classification," *Stanford University, 2016.*

[9] P.K.Suri, Ekta Walia, Er. Amit Verma, "Face Detection and Gender Detection using Principle Component Analysis (PCA)," *[Computer Science and Application, KUK 2Information Technology, MMU 3Electronics Communication, PTU, 978-1-61284-486-2/111$26.00 ©2011 IEEE, PP 679-684.*

[10] S. Tamura, H. Kawai, H. Mitsumoto, "Male/female Identification from 8 to 6 very low-resolution face images by neural network," *Pattern Recognition Letters, vol.29, No 2, pp. 331-335,1996.*

[11] V. Khryashchev, L. Shmaglit, A. Priorov, A. Shemyakov, "Adaptive Feature Extraction for Gender Classification of Human Faces," *Proc. Of the 23rd International Conference on Computer Graphics and Vision, GraphiCon'2013*, Vladivostok, 2013. P. 71-74.

[12] Ryotatsu Iga, Kyoko Izumi, Hisanori Hayashi, Gentaro Fukano and Testsuya Ohtani, "Gender and Age Estimation from Face Images*", International Conference on The Society of Instrument and Control Engineering, pp. 756-761, August, 2003.*

[13] Ramesha Ketal, "Feature Extraction based Face Recognition, Gender and Age Classification," *International Journal on Computer Science of Engineering, Vol. 02, No. 01s, 14-23, 2010.*

[14] Md. Hafizur Rahman, Suman Chowdhury, Md. Abul Bashar "An Automatic Face Detection and Gender Identification from Color Images using Support Vector Machine" *Journal of Emerging Trends in Computing and Information Sciences Vol. 4, No. 1 Jan 2013.*

[15] B. Moghaddam and M.-H. Yang, "Gender Classification with Support Vector Machine," *Proc. Int'l Conf. Automatic Face and Gesture Recognition, pp. 306-311, Mar.2000.*

[16] https://www.dexterindustries.com/howto/use-google-cloud-vision-on-the-raspberry-pi/

[17] https://in.mathworks.com/solutions/machine-learning.html

[18] Open Source Computer Vision Library, http://sourceforge.net/projects/opencvlibrary/.

[19] Google Cloud Platform for Machine Learning https://cloud.google.com/vision/