JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Gender Determination Using Facial Recognition System

Amrit Raj Shankar
Apex Institute of Technology(CSE)
Chandigarh University
Mohali, India
20BCS4602@cuchd.in

Joydeep Chandra
Apex Institute of Technology(CSE)
Chandigarh University
Mohali, India
20BCS4604@cuchd.in

Vikram Kumar Jha
Apex Institute of Technology(CSE)
Chandigarh University
Mohali, India
20BCS4583@cuchd.in

Smita Shinde
Apex Institute of Technology(CSE)
Chandigarh University
Mohali, India
smitash3011@gmail.com

Annu
Apex Institute of Technology(CSE)
Chandigarh University
Mohali, India
20BCS4608@cuchd.in

Abstract— **Face detection is one of the most widely used computer vision applications. It is a fundamental problem in computer vision and pattern recognition. In the last decade, multiple face feature detection methods have been introduced. In recent years, the success of deep learning and convolutional neural networks (CNN) have recently shown great results in powering highly-accurate face detection solutions.**

Face detection is a computer technology that determines the location and size of a human face in digital images. Given an image, the goal of facial recognition is to determine whether there are any faces and return the bounding box of each detected face (see object detection).

Face detection is the necessary first step for all facial analysis algorithms, including face alignment, face recognition, face verification, and face parsing. Also, facial recognition is used in multiple areas such as content-based image retrieval, video coding, video conferencing, crowd video surveillance, and intelligent human-computer interfaces.

The detection of human faces is a difficult computer vision problem. Mainly because the human face is a dynamic object and has a high degree of variability in its appearance. In recent years, facial recognition techniques have achieved significant progress.

Keywords—face detection, ai, ml, OpenCV TensorFlow;

I. Introduction

Face detection is a technique that identifies or locates human faces in digital images. Face detection is performed by using classifiers. A classifier is essentially an algorithm that decides whether a given image is positive(face) or negative (not a face). A classifier needs to be trained on thousands of images with and without faces. Fortunately, OpenCV already has two pre-trained face detection classifiers, which can readily be used in a program.

A typical example of face detection occurs when we take photographs through our smartphones, and it instantly detects faces in the picture. Face detection is different from Face recognition. Face detection detects merely the presence of faces in an image while facial recognition involves identifying whose face it is and show the details.

Faces convey very rich information that is critical for intact social interaction. To extract this information efficiently, faces should be easily detected from a complex visual scene.

II. Specification

a) Software:

- 8 GB RAM
- Processor - 1.5–4.5x
- Monitor – 15.6”

b) Hardware:

- Python Compiler with required Libraries and Modules
- Language: Python
- Operating System: Windows 7/8/10/11
- Deep Learning
- Computer Vision
- OpenCV
- Tensorflow

III. TimeLine

- **Feb 4th Week:** Started paper work + Synopsis + Report + going through various references to dive deep in the topic at every edge + Approach the algorithm of code.

- **Mar 1st Week:** Collection and installation of relevant modules and libraries, creating the suitable environments with all command's implementation+ Paper work.
- **Mar 2nd Week:** Facing / correcting the error occurred at the implementation of modules adoring with the packages + Paper work + Change the algorithm + Begin the steps for coding.
- **Mar 3rd Week:** Work on methods used for facial detection and how to apply those methods efficiently.
- **Mar 4th Week:** Work on completion of all the paper works asked by the supervisor .
- **Apr 1st Week:** Work on main body with proper installation of modules, models, packages, libraries followed by modules importing in the code + Started main body + Model making and training.
- **Apr 2nd Week:** Modify the program, making it smarter and more modern after its model training.

- **Apr 3rd Week:** Have an overview on the functioning, model train, response + Complete paper / docs work, having a final approach of the project

IV. Literature Review

A. EXISTING SYSTEM

Numerous robust algorithms have been developed and claimed to have accurate performance to tackle face detection and recognition problems. These algorithms or methods are the most successfully and widely used for face detection and recognition applications. The algorithms are as follow:

1. Principle Component Analysis (PCA)
   a. Eigenface
2. Linear Discriminant Analysis (LDA)
   a.Fisherface
3. Skin colour based algorithm
   a. Red-Blue-Green(RBG)
   b.YCBCr (Luminance -Chrominance)
   c.Hue-Saturation Intensity
4. Wavelet based algorithm
   a.Gabor Wavelet

Artificial neural networks based algorithm :-
   a. Fast Forward
   b. Back Propagation
   c. Radial Basis Function (RBF)

No specific justification can be made as a conclusion on which algorithm is the best for specific tasks or challenge such as various databases, various poses, illumination tolerance and facial expressions variations.

The performance of the algorithms depends on numerous factors to be taken into account. Instead of using these algorithms solely, they can be improved or enhanced to become a new method or hybrid method that yields a better performance.

Poly-NL: Linear Complexity Non-local Layers with Polynomials

Spatial self-attention layers, in the form of Non-Local blocks, introduce long-range dependencies in Convolutional Neural Networks by computing pairwise similarities among all possible positions. Such pairwise functions underpin the effectiveness of non-local layers, but also determine a complexity that scales quadratically with respect to the input size both in space and time. This is a severely limiting factor that practically hinders the applicability of non-local blocks to even moderately sized inputs. Previous works focused on reducing the complexity by modifying the underlying matrix operations, however in this work we aim to retain full expressiveness of non-local layers while keeping complexity linear. We overcome the efficiency limitation of non-local blocks by framing them as special cases of 3rd order polynomial functions. This fact enables us to formulate novel fast Non-Local blocks, capable of reducing the complexity from quadratic to linear with no loss in performance, by replacing any direct computation of pairwise similarities with element-wise multiplications. The proposed method, which we dub as "Poly-NL", is competitive with state-of-the-art performance across image recognition, instance segmentation, and face detection tasks, while having considerably less computational overhead.

RetinaFace: Single-stage Dense Face Localisation in the Wild

Though tremendous strides have been made in uncontrolled face detection, accurate and efficient face localisation in the wild remains an open challenge. This paper presents a robust single-stage face detector, named RetinaFace, which performs pixel-wise face localisation on various scales of faces by taking advantages of joint extra-supervised and self-supervised multi-task learning. Specifically, We make contributions in the following five aspects: (1) We manually annotate five facial landmarks on the WIDER FACE dataset and observe significant improvement in hard face detection with the assistance of this extra supervision signal. (2) We further add a self-supervised mesh decoder branch for predicting a pixel-wise 3D shape face information in parallel with the existing supervised branches. (3) On the WIDER FACE hard test set, RetinaFace outperforms the state of the art average precision (AP) by 1.1% (achieving AP equal to 91.4%). (4) On the IJB-C test set, RetinaFace enables state of the art methods (ArcFace) to improve their results in face verification (TAR=89.59% for FAR=1e-6). (5) By employing light-weight backbone networks, RetinaFace can run real-time on a single CPU core for a VGA-resolution image.

DSFD: Dual Shot Face Detector

In this paper, we propose a novel face detection network with three novel contributions that address three key aspects of face detection, including better feature learning, progressive loss design and anchor assign based data augmentation, respectively. First, we propose a Feature Enhance Module (FEM) for enhancing the original feature maps to extend the single shot detector to dual shot detector. Second, we adopt Progressive Anchor Loss (PAL) computed by two different sets of anchors to effectively facilitate the features. Third, we use an Improved Anchor Matching (IAM) by integrating novel anchor assign strategy into data augmentation to provide better initialization for the regressor. Since these techniques are all related to the two-stream design, we name the proposed network as Dual Shot Face Detector (DSFD). Extensive experiments on popular benchmarks, WIDER FACE and FDDB, demonstrate the superiority of DSFD over the state-of-the-art face detectors.

Selective Refinement Network for High Performance Face Detection

High performance face detection remains a very challenging problem, especially when there exists many tiny faces. This paper presents a novel single-shot face detector, named Selective Refinement Network (SRN), which introduces novel two-step classification and regression operations selectively into an anchor-based face detector to reduce false positives and improve location accuracy simultaneously. In particular, the SRN consists of two modules: the Selective Two-step Classification (STC) module and the Selective Two-step Regression (STR) module. The STC aims to filter out most simple negative anchors from low level detection layers to reduce the search space for the subsequent classifier, while the STR is designed to coarsely adjust the locations and sizes of anchors from high level detection layers to provide better initialization for the subsequent regressor. Moreover, we design a Receptive Field Enhancement (RFE) block to provide more diverse receptive field, which helps to better capture faces in some extreme poses. As a consequence, the proposed SRN detector achieves state-of-the-art performance on all the widely used face detection benchmarks, including AFW, PASCAL face, FDDB, and WIDER FACE datasets. Codes will be released to facilitate further studies on the face detection problem.

HPRNet: Hierarchical Point Regression for Whole-Body Human Pose Estimation

In this paper, we present a new bottom-up one-stage method for whole-body pose estimation, which we call "hierarchical point regression," or HPRNet for short. In standard body pose estimation, the locations of ∼17 major joints on the human body are estimated. Differently, in whole-body pose estimation, the locations of fine-grained keypoints (68 on face, 21 on each hand and 3 on each foot) are estimated as well, which creates a scale variance problem that needs to be addressed. To handle the scale variance among different body parts, we build a hierarchical point representation of body parts and jointly regress them. The relative locations of fine-grained keypoints in each part (e.g. face) are regressed in reference to the center of that part, whose location itself is estimated relative to the person center. In addition, unlike the existing two-stage methods, our method predicts whole-body pose in a constant time independent of the number of people in an image. On the COCO WholeBody dataset, HPRNet significantly outperforms all previous bottom-up methods on the keypoint detection of all whole-body parts (i.e. body, foot, face and hand); it also achieves state-of-the-art results on face (75.4 AP) and hand (50.4 AP) keypoint detection.

Condensation-Net: Memory-Efficient Network Architecture with Cross-Channel Pooling Layers and Virtual Feature Maps

"Lightweight convolutional neural networks" is an important research topic in the field of embedded vision. To implement image recognition tasks on a resource-limited hardware platform, it is necessary to reduce the memory size and the computational cost. The contribution of this paper is stated as follows. First, we propose an algorithm to process a specific network architecture (Condensation-Net) without increasing the maximum memory storage for feature maps. The architecture for virtual feature maps saves 26.5% of memory bandwidth by calculating the results of cross-channel pooling before storing the feature map into the memory. Second, we show that cross-channel pooling can improve the accuracy of object detection tasks, such as face detection, because it increases the number of filter weights. Compared with Tiny-YOLOv2, the improvement of accuracy is 2.0% for quantized networks and 1.5% for full-precision networks when the false-positive rate is 0.1. Last but not the least, the analysis results show that the overhead to support the cross-channel pooling with the proposed hardware architecture is negligible small. The extra memory cost to support Condensation-Net is 0.2% of the total size, and the extra gate count is only 1.0% of the total size.

CASSOD-Net: Cascaded and Separable Structures of Dilated Convolution for Embedded Vision Systems and Applications

The field of view (FOV) of convolutional neural networks is highly related to the accuracy of inference. Dilated convolutions are known as an effective solution to the problems which require large FOVs. However, for general-purpose hardware or dedicated hardware, it usually takes extra time to handle dilated convolutions compared with standard convolutions. In this paper, we propose a network module, Cascaded and Separable Structure of Dilated (CASSOD) Convolution, and a special hardware system to handle the CASSOD networks efficiently. A CASSOD-Net includes multiple cascaded 2×2 dilated filters, which can be used to replace the traditional 3×3 dilated filters without decreasing the accuracy of inference. Two example applications, face detection and image segmentation, are tested with dilated convolutions and the proposed CASSOD modules. The new network for face detection achieves higher accuracy than the previous work with only 47% of filter weights in the dilated convolution layers of the context module. Moreover, the proposed hardware system can accelerate the computations of dilated convolutions, and it is 2.78 times faster than traditional hardware systems when the filter size is 3×3.

Hardware Architecture of Embedded Inference Accelerator and Analysis of Algorithms for Depthwise and Large-Kernel Convolutions

In order to handle modern convolutional neural networks (CNNs) efficiently, a hardware architecture of CNN inference accelerator is proposed to handle depthwise convolutions and regular convolutions, which are both essential building blocks for embedded-computer-vision algorithms. Different from related works, the proposed architecture can support filter kernels with different sizes with high flexibility since it does not require extra costs for intra-kernel parallelism, and it can generate convolution results faster than the architecture of the related works. The experimental results show the importance of supporting depthwise convolutions and dilated convolutions with the proposed hardware architecture. In addition to depthwise convolutions with large-kernels, a new structure called DDC layer, which includes the combination of depthwise convolutions and dilated convolutions, is also analyzed in this paper. For face detection, the computational costs decrease by 30%, and the model size decreases by 20% when the DDC layers are applied to the network. For image classification, the accuracy is increased by 1% by simply replacing 3×3 filters with 5×5 filters in depthwise convolutions.

B. PROPOSED SYSTEM

Face detection is a computer vision technology that helps to locate/visualize human faces in digital images. This technique is a specific use case of object detection technology that deals with detecting instances of semantic objects of a certain class (such as humans, buildings or cars) in digital images and videos.

C. We are using tensorflow to build our face detection model.Tensorflow is very

D. popular in computer vision fields.

1) *Data Visualization*
2) *Data Augmentation*
3) *Splitting the data*
4) *Building the Model*
5) *Pre-Training the CNN model*
6) *Training the CNN model*
7) *Labeling the Information*
8) *Importing the Face detection Program*
9) *Detecting the face*

This system is very advantageous and detects the face accurately as using tensorflow and openCV is the perfect method for detecting high-end images.

V. Objectives

1) The proposed system is aimed to carry out work leading to the development of an approach for face detection.
2) Challenges in face detection are the reasons which reduce the accuracy and detection rate of facial recognition.
3) These challenges are complex background, too many faces in images, odd expressions, illuminations, less resolution, face occlusion, skin color, distance, orientation, etc.
4) Our face detection system aims to cover all the possible challenges and provide the accurate results.

VI. PROBLEM FORMULATION

The technology is growing day-by-day and still there exist many factors that have the scope of improvement. the existing systems have many cons few of which are solved an improved by our system. mentioned below are few cons of existing system and the solution that our system provides to rectify these problems of exiting system.

VII. Challenges

Unusual expression. Human faces in an image may show unexpected or odd facial expressions. Illuminations. Some image parts may have very high or low illumination or shadows.
Skin types. Detecting faces of different face colors is challenging for detection and requires a wider diversity of training images.
Distance. If the distance to the camera is too high, the object size (face size) may be too small. Orientation. The face orientation and angle toward the camera impact the rate of face detection.
Complex background. A high number of objects in a scene reduces the accuracy and rate of detection. Many faces in one image. An image with a high number of human faces is very challenging for an accurate detection rate.
Face occlusion. Faces may be partially hidden by objects such as glasses, scarves, hands, hairs, hats, and other objects, which impacts the detection rate.
Low resolution. Low-resolution images or image noise impacts the detection.

Modifiable and Customizable: The currently existing systems are written by the programmers and are the code isn't available to the users. So, if the user wants to modify anything in the current existing systems, it’s not possible for him/her.

Image-based approach Technique:
Image-based methods try to learn templates from examples in images. Hence, appearance-based methods rely on machine learning and statistical analysis techniques to find the relevant characteristics of “face” and “no-face” images. The learned characteristics are in the form of distribution models or discriminant functions that is applied for face detection tasks.
Examples: Image-based approaches include neural networks (CNN), support vector machines (SVM), or Adaboost.
Advantages: Good performance, higher efficiency

VIII. Methodology

- Face detection is a computer technology that determines the location and size of a human face in digital images.
- TensorFlow and OpenCV :- OpenCV and TensorFlow can be used together to develop and deploy computer vision applications.
- Deep learning attempts to mimic the human brain—albeit far from matching its ability—enabling systems to cluster data and make predictions with incredible accuracy.

Modules downloaded To Create Environment:

OpenCV: OpenCV is the huge open-source library for the computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today’s systems.

TensorFlow : TensorFlow is an open-source end-to-end platform for creating Machine Learning applications. It is a symbolic math library that uses dataflow and differentiable programming to perform various tasks focused on training and inference of deep neural networks.

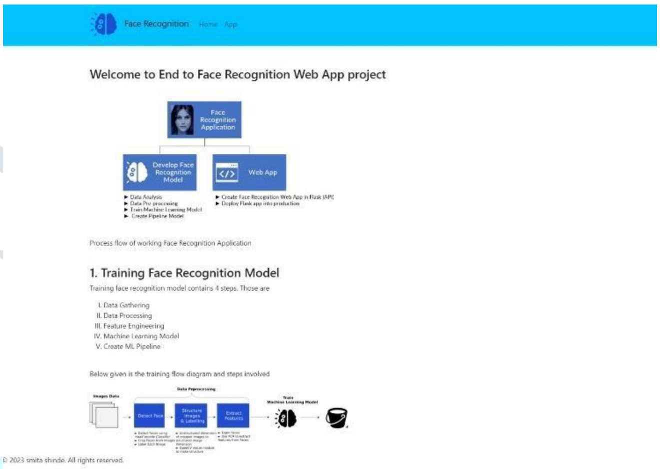
- **Process Of Building a Virtual Assistant**

STEP1:- Data Visualization
STEP2:- Data Augmentation
STEP3:- Splitting the data
STEP4:- Building the Model
STEP5:- Pre-Training the CNN model
STEP6:- Training the CNN model
STEP7:- Labeling the Information
STEP8:- Importing the Face detection Program
STEP9:- Detecting the face

IX. Applications

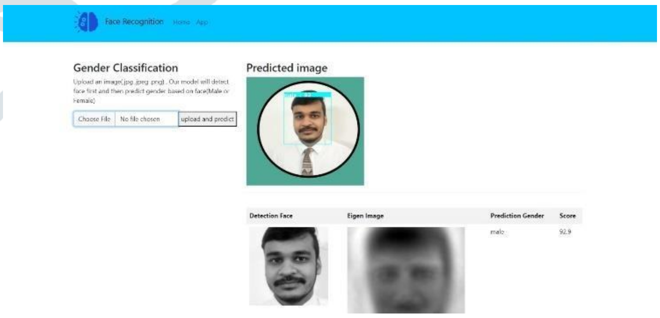
- Crowd surveillance. Face detection is used to detect and analyze crowds in frequented public or private areas.
- uman-computer interaction (HCI). Multiple human-computer interaction-based systems use facial recognition to detect the presence of humans.
- Photography. Some recent digital cameras use face detection for autofocus. Mobile apps use facial recognition to detect regions of interest in slideshows.
- Facial feature extraction. Facial features like nose, eyes, mouth, skin color and more can be extracted from images.
- Gender classification. Applications are built to detect gender information with face detection methods.
- Face recognition. A face recognition system is designed to identify and verify a person from a digital image or video frame.

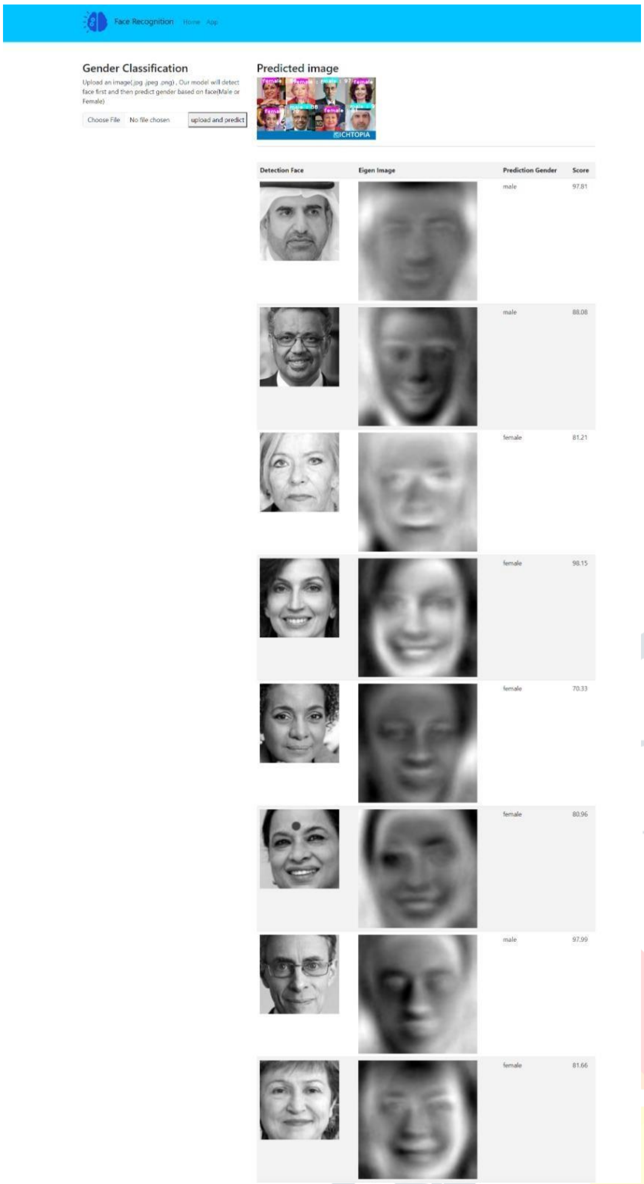
X. Result and Output



Website for execution

Face Recognition Interface





Upload photo and successfully verified

XI. Conclusion

The major finding of the work will be presented in this chapter. Also, directions for extending the current study will be discussed. The conclusion will include everything from top in brief. It contains introduction, needs, advantages, problems, scope and suggestions.

References

[1] https://ieeexplore.ieee.org/abstract/document/7515906/

[2] https://iopscience.iop.org/article/10.1088/1742- 6596/1755/1/012006/meta

[3] https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3753064

[4] https://www.researchgate.net/profile/Harshit-Nigam-3/publication/358374795_Review_of_Facial_Recognition_Technique s/links/61fe7873870587329e943c33/Review-of-Facial-Recognition-Techniques.pdf

[5] https://www.sciencedirect.com/science/article/pii/S131915781930939 5

[6] https://ieeexplore.ieee.org/abstract/document/9137850/

[7] https://www.academia.edu/download/59869714/IRJET-V6I480420190626-73439-rtrgk7.pdf

[8] https://www.mdpi.com/2078-2489/13/6/268

[9] https://ijresonline.com/archives/ijres-v7i3p115

[10] https://www.irjmets.com/uploadedfiles/paper/issue_6_june_2022/261 81/final/fin_irjmets1655416466.pdf

[11] https://www.academia.edu/download/60677306/face-recognition-system-IJERTV8IS05015020190923-25057-1sws2am.pdf

[12] https://scholar.google.com/scholar?cluster=12786622162193684980&hl=en&as_sdt=2005&sciodt=0,5

[13] https://publications.eai.eu/index.php/IoT/article/view/2346

[14] https://hal.inria.fr/hal-03620410/

[15] https://www.koreascience.or.kr/article/JAKO202020455277460.page

[16] https://arxiv.org/abs/2201.06220

[17] http://www.aimspress.com/aimspress-data/electreng/2021/1/PDF/electroneng-05-01-005.pdf