HUMAN AUTHENTICATION SYSTEM USING DEEP LEARNING TECHNIQUES

A project work report

Submitted in Partial Fulfilment for the Award of the Degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

Submitted by

G. SAI TEJA	**20B91A1255**
CH. NARASIMHA MURTHY	**20B91A1233**
CH. MANEESH GOWTHAM	**20B91A1225**
D. V. KALI VARA PRASAD	**20B91A1250**

Under the esteemed Guidance of

Dr.K. KISHORE RAJU
Associate Professor

DEPARTMENT OF INFORMATION TECHNOLOGY
S.R.K.R ENGINEERING COLLEGE (AUTONOMOUS)
(Approved by AICTE, NEW Delhi, Affiliated to JNTU University, Kakinada)
CHINNA AMIRAM: BHIMAVARAM-534202

April-2024

S.R.K.R ENGINEERING COLLEGE (A)

(Approved by AICTE, New Delhi. Affiliated to JNTU UNIVERSITY. KAKINADA)

CHINNA AMIRAM, BHIMAVARM-534204

Certificate

This is to certify that the project work entitled **“HUMAN AUTHENTICATION SYSTEM USING DEEP LEARNING TECHNIQUES”** is bonafide work submitted by G. **SAI TEJA (RegdNo:20B91A1255), CH. NARASIMHA MURTHY (RegdNo:20B91A1233), CH. MANEESH GOWTHAM (RegdNo:20B91A1225), D. V. KALI VARA PRASAD (RegdNo:20B91A1250) who** carried out the project work under my supervision in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in INFORMATION TECHNOLOGY during the academic year 2020-2024.

HOD Signature	**GUIDE Signature**
Dr.Bh.V.S.R.K. Raju	Dr.K. Kishore Raju
Professor, HOD-IT Dept	Associate Professor

CERTIFICATION OF EXAMINATION

This is to certify that I have examined the concept and hereby accord my approval of it as a project work entitled "**HUMAN AUTHENTICATION SYSTEME USING DEEP LEARNING TECHNIQUES**" carried out a presented in a manner required for its acceptance on partial fulfilments for the award of the degree of BACHELOR OF TECHNOLOGY for which it has been submitted.

This approval does not necessarily endorse or accept every statement made opinion expressed or conclusion drawn as recorded in the project report it only signifies the acceptance of the report for the purpose for which submitted

Signature

Project Guide :

Senior Faculty :

External Examiner :

HOD :

SELF DECLARARTION

We hereby declare that the project work entitled **"HUMAN AUTHENTICATION SYSTEM USING DEEP LEARNING TECHNIQUES"** is a genuine work carried out by us in B.Tech., Information Technology at SRKR Engineering College(A), Bhimavaram and has not been submitted either in part or full for the award of any other degree or diploma in any other institute or University.

Name	Registration No	Signature
GANDHAM SAI TEJA	**20B91A1255**	
CHELLEM NARASIMHA MURTHY	**20B91A1233**	
CHALLA MANEESH GOWTHAM	**20B91A1225**	
DURBHA V R L N D S S H KALI VARA PRASAD	**20B91A1250**	

ACKNOWLEDGEMENT

The victorious completion of this project would be incomplete without greeting those who made it possible and whose guidance and encouragement made efforts that taken to the success.

The thesis presented here is the work accomplished under the enviable and scholarly guidance of “**Dr. K. Kishore Raju**”, Associate Professor of Information Technology Department. We are grateful towards the indomitable support. We express deep gratitude for his inspiring remarks and helping this project.

We take this opportunity to express our sincere thanks to "**Dr. BH.V. S Rama Krishnam Raju**", Head of the Department, Information Technology for his support and encouragement.

We express our sincere thanks to "**Prof. K V Murali Krishnam Raju**", PRINCIPAL, S.R.K.R. Engineering College, Bhimavaram for giving us this opportunity for the successful completion of this project.

Last but not the least, we also express our sincere thanks to other teaching and non-teaching staff for their support in accomplishing this project.

G. SAI TEJA (20B91A1255)

CH. NARASIMHA MURTHY (20B91A1233)

CH. MANEESH GOWTHAM (20B91A1225)

D. V. KALI VARA PRASAD (20B91A1250)

ABSTRACT

This project aims to enhance security by leveraging deep learning techniques, specifically convolutional neural networks (CNNs), to verify individuals' identities in front of a camera. Upon recognizing an authorized user from a pre-established database, the system displays their details, including their name. However, if an individual is identified as unauthorized, the system utilizes pre-trained models to predict and display their age, gender, and emotion. Additionally, to alert security personnel of unauthorized attempts, the system incorporates a beep sound. This dual functionality not only strengthens security measures by restricting access to authorized users but also gathers valuable demographic data on unauthorized attempts. The project addresses the increasing demand for robust and efficient authentication systems in modern applications, ensuring heightened security and user privacy.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction about project area and Face Recognition

The field of human authentication systems is crucial for ensuring secure access to digital services and physical locations. As traditional methods evolve, there's a shift towards sophisticated approaches, including biometric authentication and machine learning. Our project is about making a smart facial recognition system using deep learning. Deep learning helps computers recognize faces by looking at photos, much like how humans do but faster and more accurately. We are using this technology to check if someone is allowed to enter a place or access something. If the system knows the person (because their information is already saved), it will quickly confirm their identity and show their details. But here's what makes our system special, it doesn't just stop at identifying people it knows. If it sees someone it doesn't recognize. Instead, it will use its smart technology to guess their age, whether they're male or female, and what kind of face expression seem to be in, and it alerts the admin by playing a Beep sound. This is helpful because it gives more information about who tried to enter and how they were feeling, which could be important for security reasons. Deep learning is a branch of machine learning which is based on artificial neural networks. It is capable of learning complex patterns and relationships within data. In deep learning, we don't need to explicitly program everything. It has become increasingly popular in recent years due to the advances in processing power and the availability of large datasets. Because it is based on artificial neural networks (ANNs) also known as deep neural networks (DNNs). These neural networks are inspired by the structure and function of the human brain's biological neurons, and they are designed to learn from large amounts of data. The key characteristic of Deep Learning is the use of deep neural networks, which have multiple layers of interconnected nodes. These networks can learn complex representations of data by discovering hierarchical patterns and features in the data. Deep Learning algorithms can automatically learn and improve from data without the need for manual feature engineering.

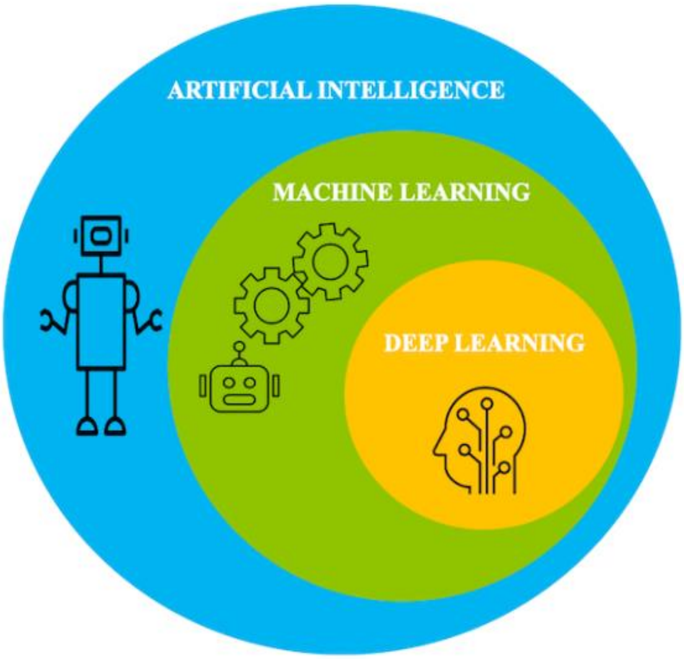


Fig 1. 1.1 Existence of Deep Learning

Deep Learning has achieved significant success in various fields, including image recognition, natural language processing, speech recognition, and recommendation systems. Some of the popular Deep Learning architectures include Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Deep Belief Networks (DBNs). Training deep neural networks typically requires a large amount of data and computational resources. However, the availability of cloud computing and the development of specialized hardware, such as Graphics Processing Units (GPUs), has made it easier to train deep neural networks.

In our project, we're using deep learning to make a smart facial recognition system. This means the computer can look at pictures of people's faces and figure out who they are. We're using a special kind of deep learning called convolutional neural networks (CNNs) to do this. These networks learn from lots of pictures of faces to recognize them accurately. Our system can also guess things about people it doesn't know. For example, it can guess how old they are, whether they're a man or a woman, and even how they're feeling. This helps make our system smart and useful for security.

Facial recognition is a technology that uses algorithms to identify or verify individuals based on their unique facial features. It works by analysing patterns in facial images, such as the distance between the eyes, nose shape, and jawline. Facial recognition systems can be used for various purposes, including security access control, surveillance, and identity verification in digital applications. These systems capture facial images from photos or video streams and compare them against a database of known faces to make a match. Advances in deep learning and artificial intelligence have greatly improved the accuracy and reliability of facial recognition technology in recent years. However, concerns about privacy, bias, and security risks remain, prompting ongoing discussions and debates about its ethical and legal implications. Despite these challenges, facial recognition continues to be adopted in various industries and applications for its convenience and efficiency in identifying individuals.

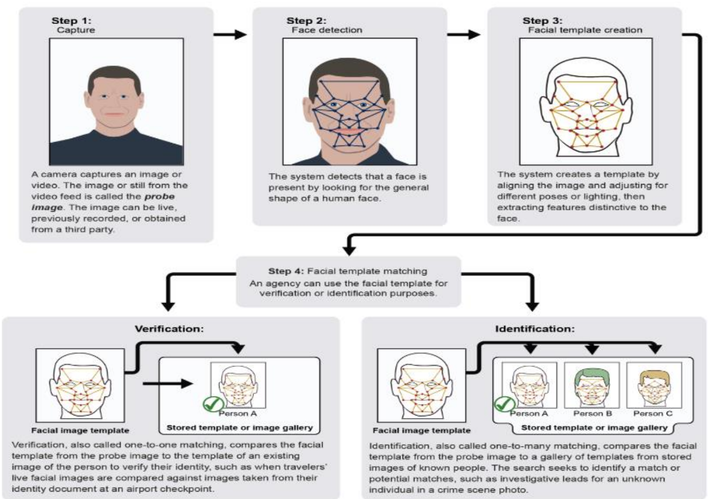


Fig 1.1.2 Face Recognition work flow

CHAPTER 2

LITERATURE SURVEY

Gil Levi et. al [1] proposed Age and Gender Classification using Convolutional Neural Networks. With this study, we're seek to close the discrepancy among automated recognition of faces capabilities and estimation of age and gender methods. To achieve this objective, we follow the successful precedence set by relevant facial identification systems: Recognizing faces addresses disclosed over the past decade demonstrate that the use of deep convolutional neural networks (CNN) may result in substantial advances [3]. We exhibit comparable advantages using a fundamental network structure intended to accommodate for the limited availability of right age and gender classifications in available face data sets.

Octavio Arriaga et. al [2] proposed Real-time Convolutional Neural Networks for Emotion and Gender Classification. The research presented here suggests and executes an overall CNN architecture platform to create an instant CNN program. The algorithms have been verified using a real-time expression detection system that identifies individuals and labels emotion in human-level reliability. They provide two alternatives and assess either based on precision of testing as well as the number of parameters. The two models were developed with an objective to achieve the best possible efficiency relative the amount of variables proportion. Following offering an overview of the instruction methodology installed, they will conduct assessments using established reference datasets. Drawbacks: Some Female subjects mistakenly classified as males and Some Male subjects mistakenly classified as females

Rapid object Detection using a Boosted cascade of Simple is proposed by Paul Viola Mitsubishi [3]. The article presents a deep learning approach to visual recognition of objects which can analyse photographs swiftly and achieve excellent identification rates. The initial step was the launch of an innovative depict description called by the term "The integral Image," that allows the detector's detection characteristics to be estimated quickly. The second approach is an AdaBoost-based approach to learning it chooses just a handful of important visual characteristics among a greater number to generate highly effective classifiers. The last modification involves combining complex classifiers in a "cascade" to remove areas of background but concentrate on potential object-like areas. Initial studies indicate a front initial experiment demonstrated that a frontal face classifier constructed from 200 features yields a detection rate of 95% with a false positive rate of 1 in 14084

Face Detection and Recognition Using OpenCV is introduced by M. Khan et. al [4]. Face Detection and Recognition Recognition of faces and image/video detection through OpenCV is an increasingly prevalent topic of informatics investigation. In real time recognition of faces is a compelling discipline that addresses a rapidly evolving problem. The principle behind the PCA (Principal Component Analysis) facial recognition algorithm is suggested here. Key component analysis (PCA) is a method of statistics that comes under the umbrella of the field of factor analysis. The Principal Component Analysis (PCA) attempts to reduce the dimension of the space of features required to effectively characterize the information beyond the present enormous amount of space. The broad 1-D pixel vector generated from a two-dimensional image of a face in compressed primary spatial features is intended for recognizing facial features via the use of the PCA. A self-space.

Lihong Wan et. al [5] proposed Recognition of faces utilizing convolutional neural network (2017), A novel approach for face recognition under two distinct circumstances has been laid out using convolutional neural networks (CNN) and subspace learning. The activation vector of the CNN architecture's fully connected layer was calculated via the feature extractor of the VGG-Face highly sophisticated CNN architecture, that has been trained on an enormous database. subsequently, two separate subdomain methods for learning, linear discriminate analysis (LDA) and whitening principal component analysis (WPCA), are presented in environments with a number of samples per dependent as well as just one sample per subject, respectively. Algorithms applied comprise convolutional neural network design and principal component analysis. Drawback: CNN does not detect object spot and orientation.

X. Ren et. al [6] introduced Face modelling method using Dlib (2017) Face modelling method using Dlib, proposes a gradient enhancement approach. It computes the real form of a global optimization model, as well as the shape of training patterns for linear least squares fitting. To accomplish autonomous localization of face feature points, the model is used to test regression estimation of sample feature point placements as well as form optimization. as well as a regress cascade learning approach. There has also been research into the relation between the regularisation parameter and the overfitting issue. Simultaneously, affine synthesizes data when there is insufficient data.

Peng et. al [7] proposed Face Recognition Technology (2020), A Brief Description of Face Recognition Technology. The method of face recognition utilizes a person's facial features to recognize them. Individuals gather their facial images, which are then eventually automatically analysed by image recognition technology. An overview of many studies related to recognizing faces is provided in this article. The paper highlights multiple phases of development for facial detection systems of all kinds. It provides face recognition databases, general assessment standards, and face identification research for scenarios from real life.

It provides an outlook on facial recognition from the future. Future study efforts should concentrate on recognition of facial features, as it offers an extensive range of intriguing prospective uses. Algorithms used: Principal component analysis Linear discriminate analysis Support vector machine Neural networks Drawbacks: It cannot recognize rotated angled and occluded faces.

From the above literature it is inferred that an integrated system is very much needed to implement effective security system for not only authentication, but also an integrated information gathering system is badly needed, when an unauthorised user is identified.

CHAPTER 3

EXISTING SYSTEM AND PROBLEM STATEMENT

3.1 Existing system

The existing facial recognition systems primarily focus on identifying and verifying individuals against a database of known faces. These systems are widely used in security applications, smartphones, and access control mechanisms. However, they often lack the capability to provide additional information about unauthorized users, such as demographic data or emotional states, such as age and emotion which could be valuable for security analysis and understanding user behaviour.

3.2 Problem statement

In the context of evolving technological advancements, there arises a need for the creation of a robust real-time face recognition system. develop a real-time face recognition system capable of identifying known individuals and providing demographic insights for unknown faces. By leveraging face recognition techniques with a tolerance mechanism, the system accurately matches faces against pre-collected data, displaying recognized names on-screen. For unrecognized faces, the system predicts age, gender, and emotion using deep learning models. Real-time processing ensures efficient video frame analysis from a webcam feed, with overlaid information showcasing recognized names and demographic predictions. Additionally, the system incorporates a buzzer alert for unidentified faces and allows for the seamless addition of new faces to the database for future recognition.

CHAPTER 4

PROPOSED SYSTEM

The proposed system is an intelligent face recognition and analysis system designed to accurately identify individuals, predict their age, gender, and emotional state in real-time using computer vision and deep learning techniques. It utilizes a combination of face recognition algorithms, pre-trained deep learning models for age, gender and emotion prediction methods.

Advantages of Proposed System:

- *Enhanced Security*: By accurately recognizing faces, the system strengthens security measures in various domains such as access control, surveillance, and authentication.
- *Real-Time Analysis*: The system provides instantaneous analysis of individuals, allowing for quick decision-making and response in security-sensitive situations.
- *Personalized Services*: In commercial settings, the system can be used to personalize customer experiences by recognizing and analysing customer demographics and emotions.
- *Efficient Resource Allocation:* By automating the identification and analysis process, the system optimizes resource allocation, reducing the need for manual intervention in tasks such as age verification or gender profiling.
- *Scalability:* The system can be easily scaled to accommodate additional functionalities or integrate with existing security systems, making it adaptable to diverse environments and requirements.
- *Improved User Experience:* With its ability to accurately predict age, gender, and emotions, the system enhances user experience by providing tailored interactions and services based on individual characteristics.

CHAPTER 5

METHODOLOGY

5.1 SYSTEM ARCHITECTURE

We proposed an architecture for “*Human Authentication System Using Deep Learning Techniques” which* is divided into four modules that are listed below.

i. *Face Detection*

ii. *Face Recognition*

iii. *Age & Gender Prediction*

iv *Emotion Prediction*

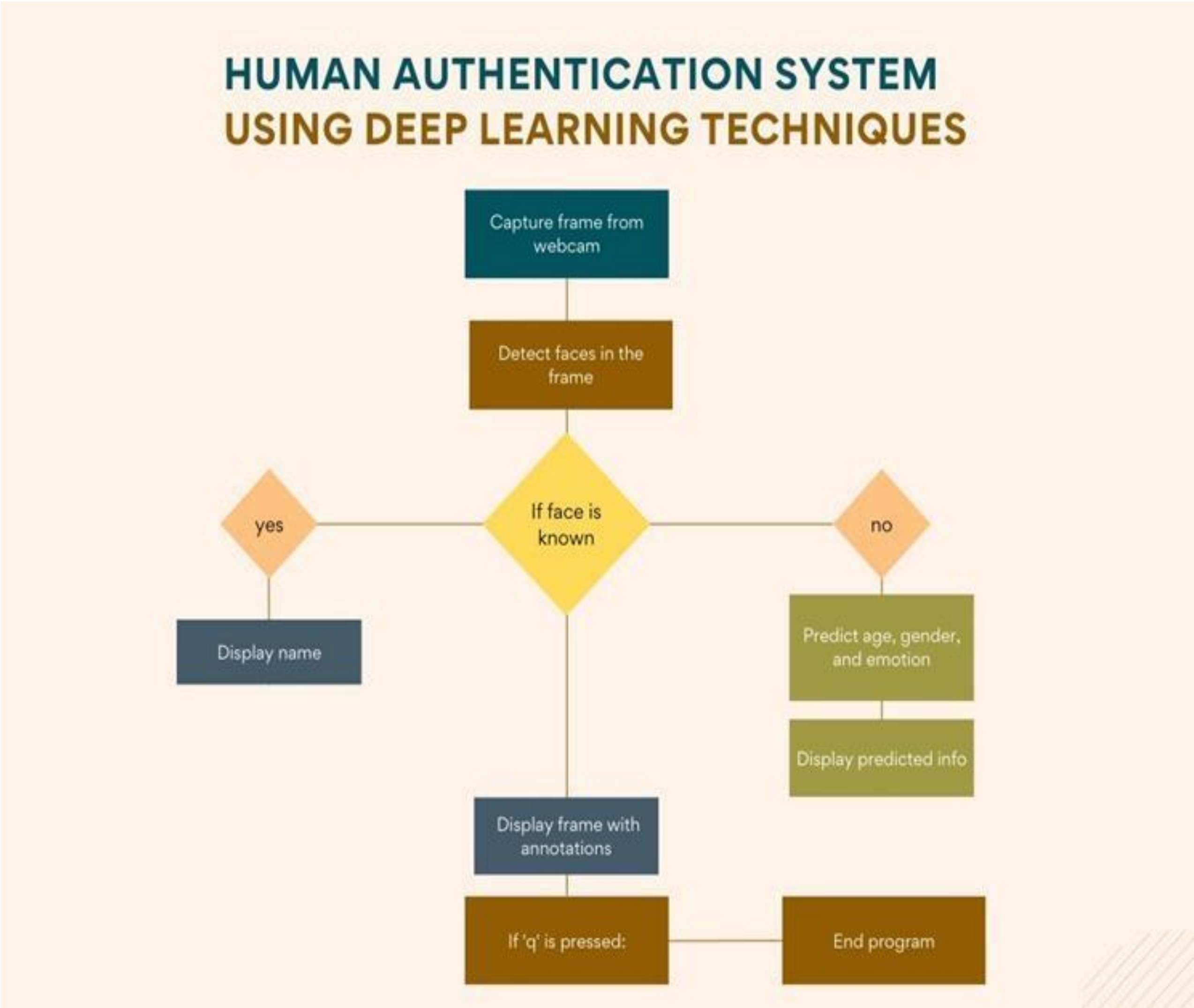


Fig. 5.1 1 System Architecture

System architecture Description:

- *Capture Frame from Webcam:* The system initiates by capturing a frame from a webcam. This frame will be used to identify the user attempting to gain access.
- *Detect Faces in the Frame:* Facial recognition software is employed to detect the presence of faces within the captured frame.
- *Face is Recognized:* If a face is recognized, the system retrieves the user's name from its database and displays it on the screen.
- *Face is Not Recognized:* Additional analysis is performed on the recognized face to predict the user's age, gender, and emotional state.
- *Display Predicted Info:* The system displays the predicted age, gender, and emotion of the recognized user.
- The process concludes when the user presses the 'q' key on the keyboard.

5.2 Proposed Methodology

Case – 1 Face Detection using Haar Cascade Classifier

Introduction:

The face detection system discussed here employs a sophisticated 38-layer cascaded classifier specifically designed to detect frontal upright faces. This section provides an overview of the methodology and technical aspects involved in training such a system.

Dataset Used:

- *Face Training Set:* The face training set consists of 4916 hand-labelled faces, meticulously scaled and aligned to a base resolution of 24 by 24 pixels. These faces were extracted from a diverse range of images obtained through a random crawl of the World Wide Web, ensuring variability in facial expressions, lighting conditions, and angles. The inclusion of a comprehensive and diverse face training set is essential for training a robust and generalizable face detection system.

- *Non-Face Subwindows:* The non-face subwindows utilized during training were sourced from 9544 images manually inspected to exclude any facial features. These images contributed approximately 350 million subwindows, serving as negative examples during the training process. The careful selection of non-face subwindows ensures the discriminative power of the classifier and minimizes false positives during face detection.

Network Architecture:

The architecture of the face detection system is characterized by a 38-layer cascaded classifier, with each layer comprising a varying number of features. The initial layers of the classifier contain a minimal number of features, gradually increasing in subsequent layers. For example, the first five layers have 1, 10, 25, 25, and 50 features respectively. These features represent simple rectangular patterns known as Haar-like features, which capture differences in pixel intensities between adjacent regions of the image. Some examples of these features include Horizontal Edge Feature, Vertical Edge Feature, Diagonal Edge Feature, Two rectangle Edge Feature, Three Rectangle Edge Feature.

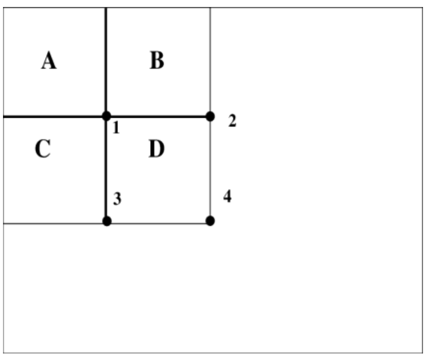


Fig 5.2.1: Extraction of Features for Object Detection

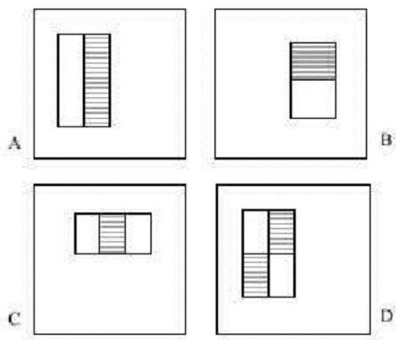


Fig 5.2.2: Visual Representation of Extracting Features for Object Detection

- Fig 5.2.1 describes the sum of the pixels within a rectangle in the integral image can be computed with four array references. The value of the integral image at location 1 is the sum of the pixels in rectangle A. The value at location 2 is $B-A$, at location 3 is $C-A$, and at location 4 is $A+D-B-C$. The sum within rectangle A can be computed as: Sum(A)=Value at 4−Value at 2−Value at 3+Value at 1
- Fig 5.2.2 The sum of the pixels which lie within the white rectangles are subtracted from the sum of pixels in the grey rectangles. Two-rectangle features are shown in (A) and (B). Figure (C) shows a three-rectangle feature, and (D) a four-rectangle feature.

Adaboost Training Procedure:

- The Adaboost training procedure is a key component of the face detection system, responsible for iteratively refining the classifier's discriminative power. Each classifier within the cascade is trained using the Adaboost algorithm, which combines multiple weak classifiers into a strong classifier. The inclusion of vertical mirror images of the training faces and the selective collection of false positives during training contribute to the robustness and accuracy of the classifier.

- The classifier cascade comprises multiple layers, each consisting of a varying number of features. The cascade is designed to efficiently detect faces while minimizing false positives, making it suitable for real-world applications. The iterative refinement process employed during training ensures the effectiveness and robustness of the cascade in detecting faces across diverse conditions.

Case – 2 FACE RECOGNITION LIBRARY:

- *Face Detection:* The library uses a pre-trained convolutional neural network (CNN) from dlib to detect faces in images. This CNN is trained to recognize faces and localize them within an image.
- *Face Encoding:* Once faces are detected, the library extracts facial features and encodes them into a numerical representation using a deep neural network. This numerical representation, often referred to as a face embedding or face descriptor, captures the unique characteristics of each face.
- *Face Recognition*: To recognize faces, the library compares the face embeddings of the detected faces with a database of known face embeddings. It uses a distance metric (typically Euclidean distance or cosine similarity) to measure the similarity between face embeddings and identify matches.
- The `face_recognition` library is built primarily using Python, leveraging the dlib library for face detection and feature extraction. Here's an overview of its key components:
- *Dlib:* The core functionality of face detection and facial feature extraction is implemented using the dlib library, which is written in C++. dlib provides highly optimized implementations of machine learning algorithms, including the CNN used for face detection.
- *Python Interface:* The `face_recognition` library provides a Python interface that wraps around the functionality provided by dlib. This allows developers to use the face recognition capabilities in Python scripts and applications without needing to directly interact with the underlying C++ code.
- *NumPy:* The library relies heavily on NumPy, a powerful library for numerical computing in Python. NumPy is used for handling arrays and

mathematical operations, which are essential for processing image data and performing computations on face embeddings.

- *Scikit-learn*: Scikit-learn, a popular machine learning library for Python, is used for various tasks such as clustering and classification. While not directly part of the core face recognition functionality, it may be used in conjunction with the library for tasks like building classifiers or clustering face embeddings.

DATASETS USED BY PRE-TRAINED MODELS:

The pre-trained models included with dlib have been trained on various datasets, including but not limited to:

- *MUCT:* The Maastricht University Computer Vision Test (MUCT) dataset contains facial images of individuals in various poses and under different lighting conditions. It's commonly used for training face detection models.
- *LFW (Labelled Faces in the Wild)* **:** The LFW dataset is a collection of face images gathered from the web. It's widely used for face recognition research and benchmarking.
- *IMDB-WIKI*: The IMDB-WIKI dataset contains face images of celebrities extracted from IMDb and Wikipedia. It's often used for training deep learning models for face recognition.
- *CASIA-WebFace*: The CASIA-WebFace dataset consists of face images collected from the web. It's commonly used for training deep learning-based face recognition models.

These datasets provide the necessary training data to teach the models how to detect faces accurately and extract meaningful facial features. The pre-trained models included with dlib have been trained on a combination of these datasets to achieve good performance across a wide range of real-world scenarios. When using the “face_recognition” library, you typically don't need to worry about obtaining or training on these datasets yourself. Instead, you can directly use the pre-trained models provided by dlib, which have already been trained on these datasets and are included with the library.

5.3 MATHEMATICAL EXPRESSIONS:

- *Euclidean Distance*: Used to measure the similarity between two feature vectors x and y.

Euclidean Distance $(x, y) = \sqrt\sum_i=1^n (x_i - y_i) ^2$

where n is the dimensionality of the feature vectors.

- *Cosine Similarity*: Another measure of similarity between two feature vectors x and y, often used in high-dimensional spaces.

Cosine Similarity $(x, y) = x \cdot y\|x\| \|y\|$

where $x \cdot y$ denotes the dot product of x and y, and $\|x\|$ and $\|y\|$ represent

the Euclidean norms of x and y respectively

- *Sigmoid Function*: Commonly used in neural networks as an activation function.

$\Sigma(x) = 11 + e^-x$

where e is Euler's number (approximately 2.71828).

- *Convolution Operation*: Fundamental operation in convolutional neural networks (CNNs) for feature extraction.

$(x * w)(i) = \sum_k=1^m x(k) \cdot w(i - k)$

where x is the input signal, w is the filter (also known as kernel), and m is the length of the filter.

- *Loss Functions*: Used in training machine learning models to quantify the difference between predicted and actual values.

- *Mean Squared Error (MSE):*

  MSE (y_true, y_pred) = 1n \sum_i=1^n (y_true^(i) - y_pred^(i)) ^2

- *Cross-Entropy Loss (Binary Classification):*

  CE(y_true, y_pred) = -1n \sum_i=1^n \left(y_true^(i) \log(y_pred^(i)) + (1 - y_true^(i)) \log(1 - y_pred^(i)) \right)

5.4 Using Pretrained Models for Age and Gender Prediction

*Introduction***:** Age and gender prediction play crucial roles in our project to identify and provide the information about Unknown persons. We utilize both the "age_net.caffemodel" and "gender_net.caffemodel" pretrained models along with their ("deploy_age.prototxt" and "gender_deploy.prototxt") to predict the age and gender of individuals from images. These pretrained models have been trained on the Adience benchmark dataset, designed for age and gender classification tasks. Our implementation is based on the Caffe deep learning framework, known for its efficiency and ease of use in deploying pre-trained models for image classification tasks.

These pretrained models have been trained on the Adience benchmark dataset, designed for age and gender classification tasks. Our implementation is based on the Caffe deep learning framework, known for its efficiency and ease of use in deploying pre-trained models for image classification tasks.

Dataset Used: The Adience dataset consists of approximately 26,000 images of 2,284 subjects. These images were collected from online sources, reflecting real-world conditions with highly unconstrained viewing conditions. Each image is annotated with age and gender labels. We utilize a five-fold, subject-exclusive cross-validation protocol for age and gender classification tasks, ensuring robust evaluation of the models' performance.

CNN Architecture:

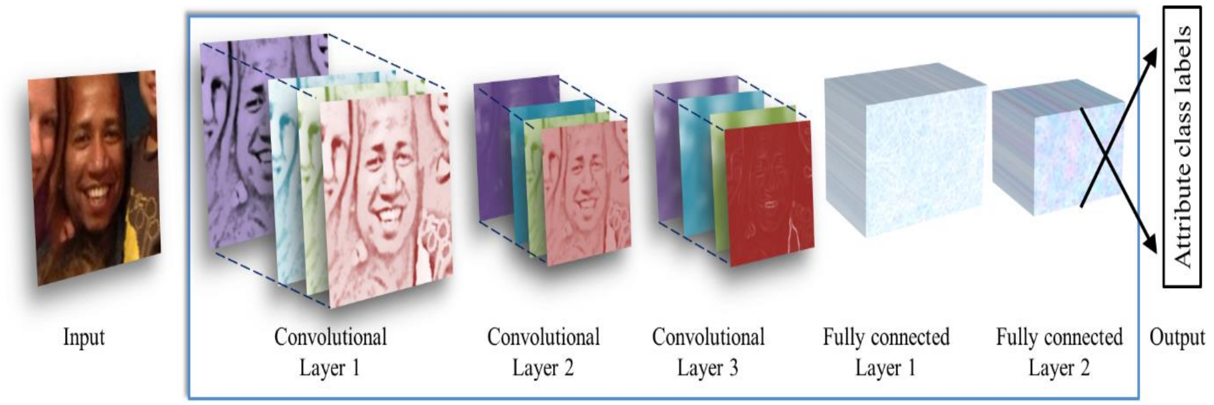


Fig 5.4.1: Illustration of CNN Architecture

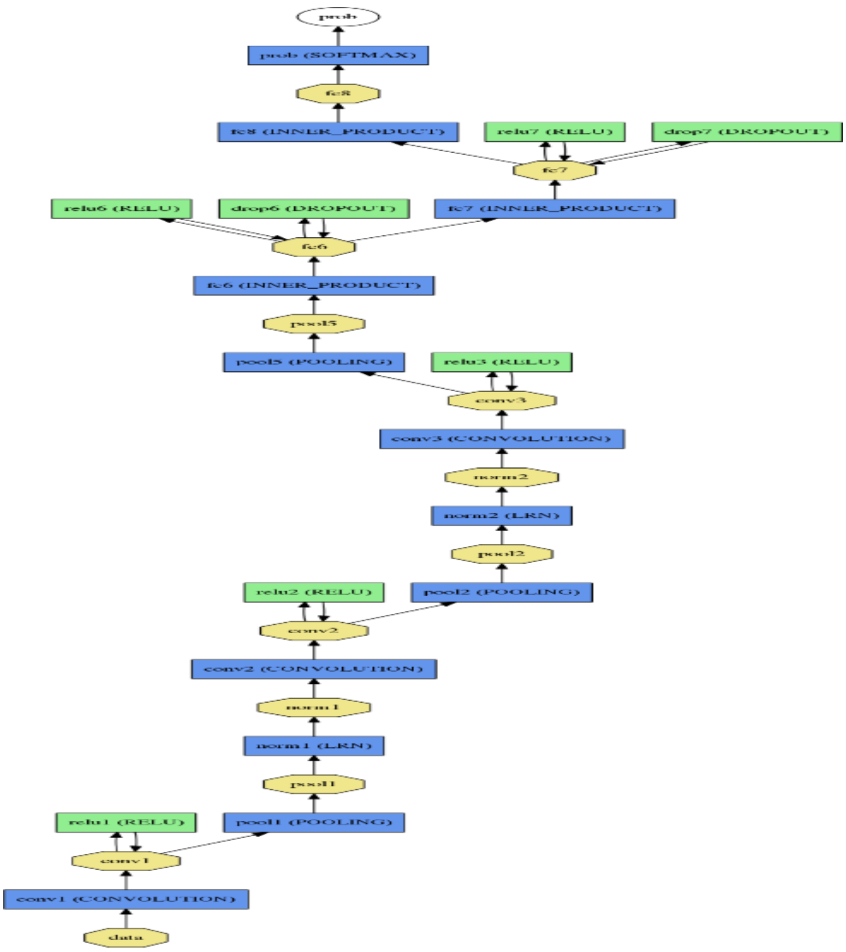


Fig 5.4.2 network architecture

Our proposed network architecture is used throughout our experiments for both age and gender classification. A more detailed, schematic diagram of the entire network design is additionally provided in Figure. The network comprises of only three convolutional layers and two fully- connected layers with a small number of neurons. This, by comparison to the much larger architectures applied. Our choice of a smaller network design is motivated both from our desire to reduce the risk of overfitting as well as the nature.

All three-color channels are processed directly by the network**.** Images are first rescaled to 256 × 256 and a crop of 227 × 227 is fed to the network**.** The three subsequent convolutional layers are then defined as follows.

Fig: **Full schematic diagram of network architecture**

- 96 filters of size 3×7×7 pixels are applied to the input in the first convolutional layer, followed by a rectified linear operator (ReLU), a max pooling layer taking the maximal value of 3 × 3 regions with two-pixel strides and a local response normalization layer.
- The 96 × 28 × 28 output of the previous layer is then processed by the second convolutional layer, containing 256 filters of size 96 × 5 × 5 pixels. Again, this is followed by ReLU, a max pooling layer and a local response normalization layer with the same hyper parameters as before.
- Finally, the third and last convolutional layer operates on the 256 × 14 × 14 blob by applying a set of 384 filters of size 256 × 3 × 3 pixels, followed by ReLU and a max pooling layer.

The following fully connected layers are then defined by:

- A first fully connected layer that receives the output of the third convolutional layer and contains 512 neurons, followed by a ReLU and a dropout layer.

- A second fully connected layer that receives the 512- dimensional output of the first fully connected layer and again contains 512 neurons, followed by a ReLU and a dropout layer.
- Finally which maps to the final classes for age or gender. 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.

Networking training: Aside from our use of a lean network architecture, we apply two additional methods to further limit the risk of overfitting. First, we apply dropout learning (i.e. randomly setting the output value of network neurons to zero). The network includes two dropout layers with a dropout ratio of 0.5 (50% chance of setting a neuron's output value to zero). Second, we use data-augmentation by taking a random crop of 227 × 227 pixels from the 256 × 256 input image and randomly mirror it in each forward-backward training pass. This similarly, to the multiple crop and mirror variations used by. Training itself is performed using stochastic gradient decent with image batch size of fifty images. The initial learning rate is e−3 reduced to e −4 after 10K iterations.

	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: Breakdown into the different Age and Gender classes of Adience Faces benchmark.

Method	**Accuracy**
Best from [10]	77.8 ± 1.3
Best from [23]	79.3 ± 0.0
Proposed using single crop	85.9 ± 1.4
Proposed using over-sample	**86.8 ± 1.4**

Table 2: Gender estimation result Listed are the mean accuracy ±standard error over all age categories.

5.5 Emotion Prediction:

In our project, the detection of human emotions plays a pivotal role to get the demographic information about the unknown person. To achieve this, we leverage a pretrained model stored in the "emotion_model.hdf5" file. This model has been meticulously trained on the FER-2013 emotion dataset, a comprehensive collection of facial images annotated with various emotional labels which encompasses emotions such as anger, disgust, fear, happiness, sadness, surprise, and neutrality. Through our utilization of the "emotion_model.hdf5" pretrained model, we endeavour to overcome these challenges and achieve accurate and nuanced emotion detection capabilities in our system.

Data set Used: FER-2013 Dataset.

This model utilized the FER-2013 dataset for emotion classification. This dataset consists of 35,887 grayscale images, each labelled with one of seven emotion classes: "angry," "disgust," "fear," "happy," "sad," "surprise," and "neutral." The dataset provides a diverse range of facial expressions captured in real-world scenarios, enabling robust training and evaluation of emotion prediction models.

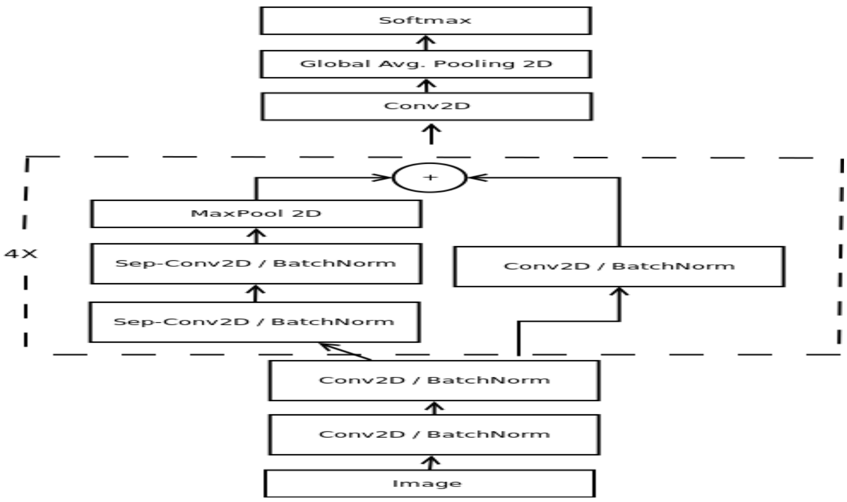


Fig 5.5.1 Illustration of Emotion prediction Network Architecture

The emotion prediction model is based on a modified CNN architecture inspired by the Xception architecture. It comprises several convolutional layers followed by batch normalization and ReLU activation functions. The specific architecture details include:

- *Number of Convolutional Layers*: The model consists of 4 residual depth-wise separable convolutional layers. These layers are designed to capture hierarchical features effectively while minimizing computational complexity. This helps in addressing the vanishing gradient problem and enables the network to learn more complex features effectively.
- *Number of Fully Connected Layers*: Fully connected layers are not used in the model architecture to reduce the number of parameters and computational burden. Instead, global average pooling is employed to summarize feature maps and produce predictions.
- *Reasoning for the Architecture*: The chosen architecture strikes a balance between model complexity, computational efficiency, and performance. The use of depth-wise separable convolutions and elimination of fully connected layers contribute to a streamlined architecture that achieves high accuracy in emotion prediction tasks while maintaining real-time performance.
- Finally, we presented a visualization of the learned features in the CNN using the guided back-propagation visualization. This visualization technique is able to show us the high-level features learned by our models and discuss their interpretability.

CHAPTER 6

IMPLEMENTATIONS

The system's implementation is founded on real-time facial attribute recognition and identification. It employs the OpenCV library's Haar cascade classifier for accurate face detection across diverse environments, ensuring reliable performance. This initial step serves as the foundation for subsequent attribute recognition processes.

Integration of pre-trained deep learning models forms a crucial component of the system. These models, encompassing gender prediction, emotion recognition, and age estimation, facilitate accurate attribute analysis in real-time. Seamless integration of these models enables the system to provide comprehensive insights into recognized individuals, thereby enhancing situational awareness.

Continuous frame capture from the webcam stream enables dynamic processing of facial attributes, with results promptly displayed to the user interface. Additionally, the implementation includes a configurable tolerance parameter for facial recognition, offering flexibility in adjusting matching thresholds to meet specific application needs. Performance optimization measures ensure efficient real-time processing, even on devices with limited computational resources, culminating in a responsive and user-friendly solution for facial attribute recognition and identification.

```
[8]: #final working code
import cv2
import numpy as np
import face_recognition
import os
import winsound  # For playing sound on Windows
from tensorflow.keras.models import load_model
```

cv2 is a Python module that provides computer vision functionalities, including image and video processing, drawing functions, and object detection.

NumPy library with the alias np, which is commonly used for numerical computations in Python. face_recognition library, which provides functionalities for face detection and recognition.

os module, which provides functions for interacting with the operating system, such as file operations.

win sound module, which allows playing sound on Windows systems.

load_model function from the model's module of TensorFlow Kera's, which is used to load pre-trained neural network models

Loading Pretrained models.

```
[ ]: # Load pre-trained models
face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade_frontalface_default.xml')
gender_model = cv2.dnn.readNet(r"C:\Users\kaliv\Downloads\ikkada\deploy_gender.prototxt", r"C:\Users\kaliv\Downloads\ikkada\gender_net.caffemodel")
emotion_model = load_model(r"C:\Users\kaliv\Downloads\ikkada\emotion_model.hdf5")

# Load the pre-trained age prediction model
age_net = cv2.dnn.readNetFromCaffe(
    r'C:\Users\kaliv\Downloads\ikkada\deploy_age.prototxt',
    r'C:\Users\kaliv\Downloads\ikkada\age_net.caffemodel')
```

These lines initialize pre-trained models essential for various tasks in computer vision. The face cascade classifier is used for face detection, while the gender recognition model predicts genders. Additionally, the emotion recognition model determines emotions, and the age prediction model estimates ages. These models collectively enable the analysis of facial attributes in real-time video streams.

*Function to predict age***.**

```
# Function to predict age
def predict_age(face_img):
    blob = cv2.dnn.blobFromImage(face_img, 1, (227, 227), (78.4263377603, 87.7689143744, 114.895847746), swapRB=False)
    age_net.setInput(blob)
    age_preds = age_net.forward()
    print("Age predictions:", age_preds)  # Add this line for debugging
    age_index = age_preds[0].argmax()
    age = age_list[age_index]
    return age
```

This function predicts the age of a given face image. It preprocesses the image into a blob compatible with the age prediction model. Then, it feeds the blob into the model and retrieves age predictions. The function prints the age predictions for debugging. Finally, it selects the age with the highest probability and returns it as the predicted age.

Function to predict Gender

```
def predict_gender(face):
    blob = cv2.dnn.blobFromImage(face, 1, (227, 227), (78.4263377603, 87.7689143744, 114.895847746), swapRB=False)
    gender_model.setInput(blob)
    gender = "Male" if gender_model.forward()[0][0] > 0.5 else "Female"
    return gender


# Function to predict emotion
```

This function predicts the gender of a given face image. It first converts the face image into a blob using the *cv2.dnn.blobFromImage* function, which preprocesses the image for input to the neural network. Then, it sets the blob as input to the gender recognition model using *gender_model.setInput()*. After that, it forwards the input through the model using *gender_model.forward ()* and compares the output probability. If the probability of being male is greater than 0.5, it predicts the gender as "Male"; otherwise, it predicts "Female". Finally, it returns the predicted gender.

Function to predict emotion

```
# Function to predict emotion
def predict_emotion(face):
    face = cv2.cvtColor(face, cv2.COLOR_BGR2GRAY)
    face = cv2.resize(face, (64, 64))  # Resize to (64, 64)
    face = face.astype("float") / 255.0
    face = np.expand_dims(face, axis=0)
    face = np.expand_dims(face, axis=-1)

    emotion_labels = ['Angry', 'Disgust', 'Fear', 'Happy', 'Sad', 'Surprise', 'Neutral']
    prediction = emotion_model.predict(face)
    return emotion_labels[np.argmax(prediction)]
```

This function predicts the emotion of a given face image. Initially, it converts the color image into grayscale using *cv2.cvtColor()* and resizes it to a fixed size of (64, 64). Then, it normalizes the pixel values to the range [0, 1]. Next, it expands the dimensions of the image to fit the expected input shape of the emotion recognition model. The function utilizes a predefined list of emotion labels and predicts the emotion label corresponding to the highest probability using the emotion recognition model. Finally, it returns the predicted emotion label.

Collect known faces from images

```python
# Function to collect known faces and their names
def collect_known_faces():
    known_face_encodings = []
    known_face_names = []

    # Example: Collect known faces from images
    known_faces_dir = r"C:\Users\kaliv\Downloads\ikkada\known_faces\v3"
    for person_dir in os.listdir(known_faces_dir):
        person_path = os.path.join(known_faces_dir, person_dir)
        if os.path.isdir(person_path):
            for filename in os.listdir(person_path):
                image_path = os.path.join(person_path, filename)
                if os.path.isfile(image_path):
                    image = face_recognition.load_image_file(image_path)
                    face_encodings = face_recognition.face_encodings(image)
                    if len(face_encodings) > 0:
                        encoding = face_encodings[0]
                        known_face_encodings.append(encoding)
                        known_face_names.append(person_dir)

    return known_face_encodings, known_face_names
```

The function *collect_known_faces ()* iterates through a directory containing images of known faces. For each person's folder, it reads image files and detects faces using the *face_recognition* library. If a face is detected in an image, its encoding is extracted and converted into a tuple. This tuple is then used as a key in a dictionary, with the person's name being the corresponding value. After iterating through all images and folders, the function returns this dictionary, mapping face encodings to their respective persons' names.

Capture video from webcam

cap = cv2.VideoCapture(0)

In this snippet, a video capture object *cap* is initialized to capture frames from the default webcam (index 0). The desired width and height for resizing frames are set to 640 and 480, respectively. Additionally, the *tolerance* variable is defined with a value of 0.5, representing the threshold for face recognition. This tolerance value determines the level of similarity required to consider a detected face as a match with a known face.

Facial Feature Recognition and Real-Time Identification

```python
while True:
    ret, frame = cap.read()
    if not ret:
        break

    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5, minSize=(30, 30))

    for (x, y, w, h) in faces:
        face = frame[y:y+h, x:x+w]

        # Encode the face for recognition
        face_encodings = face_recognition.face_encodings(frame, [(y, x+w, y+h, x)])

        if len(face_encodings) > 0:
            # Face recognized
            for face_encoding in face_encodings:
                # Compare face encoding with known face encodings
                matches = face_recognition.compare_faces(known_face_encodings, face_encoding, tolerance=tolerance)

                if True in matches:
                    # Known face recognized, get the name
                    first_match_index = matches.index(True)
                    name = known_face_names[first_match_index]

                    # Display name
                    cv2.putText(frame, f'Name: {name}', (x, y+h+20), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 255, 0), 2)
                    break
                else:
                    # Unknown face, set name as "Unknown"
                    name = "Unknown"

                    # Predict age, gender, and emotion
                    age = predict_age(face)
                    gender = predict_gender(face)
                    emotion = predict_emotion(face)

                    # Display name, age, gender, and emotion in red
                    cv2.putText(frame, f'Name: {name}', (x, y+h+20), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
                    cv2.putText(frame, f'Age: {age}', (x, y+h+50), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
                    cv2.putText(frame, f'Gender: {gender}', (x, y+h+80), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
                    cv2.putText(frame, f'Emotion: {emotion}', (x, y+h+110), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

                    # Play buzzer sound
                    winsound.Beep(1000, 500)  # Frequency: 1000Hz, Duration: 500ms

        cv2.rectangle(frame, (x, y), (x+w, y+h), (255, 0, 0), 2)
```

This code snippet continuously captures frames from the webcam using the *cap.read()* function within a *while True* loop. It converts each frame to grayscale and detects faces using the pre-trained Haar cascade classifier (*face_cascade*). For each detected face, it extracts the region of interest (ROI) and encodes it for recognition using the *face_recognition* library. If the encoding matches any known faces within a certain tolerance level, the recognized person's name is displayed on the frame. Otherwise, it predicts the person's age, gender, and emotion using separate functions *(predict_age, predict_gender, predict_emotion)* and displays them along with the label "Unknown" in red. Additionally, a buzzer sound is played when an unknown face is detected. Finally, rectangles are drawn around the detected faces on the frame.

CHAPTER 7

7.1 RESULT ANALYSIS

To test the performance of proposed system, all the modules are implemented using python 3.8.1 and is executed using Jupyter as IDE on desktop system with 16GB RAM and i7 processor with 4.6GHz processor speed. Proposed Human Authentication system captures the user video through web cam with 1080 pixels. The captured video is divided into sequence of frames and are supplied as input to the proposed architecture. The main three stages of execution of Proposed system is as follows:

- Initially. Capture faces from web cam
- Then, load every individual authorised user image(s) in separate directory/folder as shown in Fig 7.1.1.
- Finally, Captured images given as input to the system
  - If face is recognised as authorised system will display his/her name as shown in Fig 7.1.2.
  - If Not Authorised user
    - System will display Age Gender and Emotion of the User as shown in Fig 7.1.3.

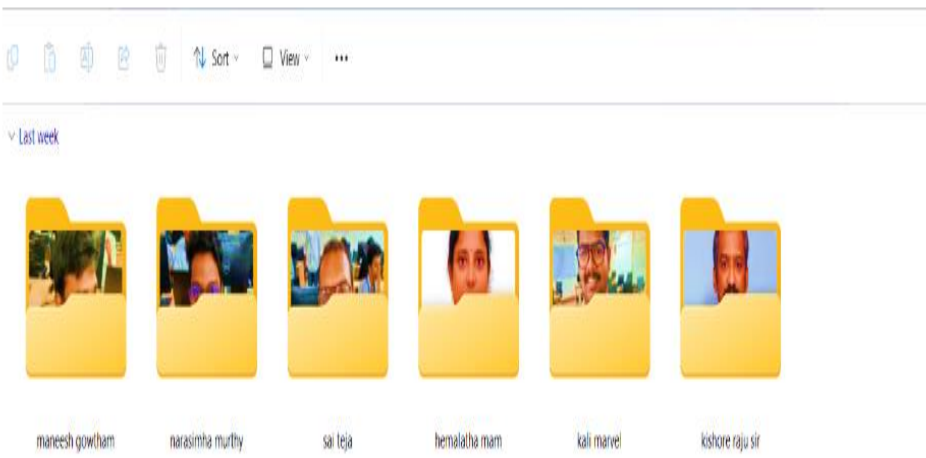


Fig 7.1.1: Existing authorised users images stored in Folders

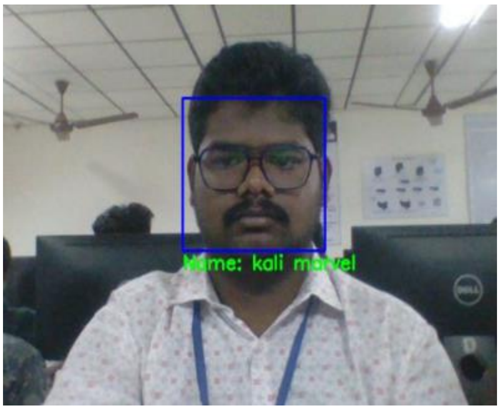


Fig 7.1.2: Recognition of authorised user

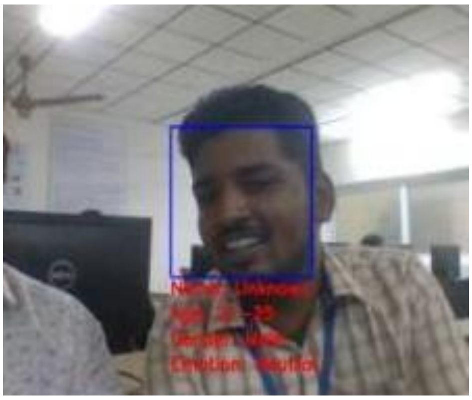


Fig 7.1.3: Recognition of Unauthorised user

System Modules	Accuracy	Recall	Precision	F1 Score
Face Recognition	90%	90%	90%	90%
Age Prediction	86%	89%	94%	93%
Gender Prediction	95%	95%	95%	94%
Emotion Prediction	94.4%	71%	73%	75%
Average	**91.35%**	**86.25%**	**88%**	**88%**

Table 3 : Performance metrics of different modules in proposed method

After conducting experments, it is evident that the face recognition achieved an accuracy rate of 90% using the dlib library , The reduction of accuracy by 10% is because of hardware limitations such as CPU and GPU effecting the accuracy of the face recognition process. The age and gender prediction both uses the same CNN algorithm as discussed in phase-4 in section-3, it was found that both Age and Gender performance is between the range of 86-95% as shown in the Table 3 and the defficiency in this algorithm is may be impacted by using few convolutional layered architecture of the pre-trained model [Fig 5.4.1] as it doesn't extract required features. For emotion prediction , it uses Guided back-propagation visualization of mini-Xception model and it produces 94.4% of accuracy and the major reductions are come across other metrics. It is concluded that the overall accuracy[91.35%] of this proposed system as mentioned in the Table 3.

Calculations :

ACCURACY TABLE FOR FACE RECOGNITION:

Actual\Predicted	Positive (Identified)	Negative (Not Identified)
Positive (Actual)	45 (TP)	5 (FN)
Negative (Actual)	5 (FP)	45 (TN)

With this confusion matrix:

- True Positives (TP) = 45
- False Negatives (FN) = 5
- False Positives (FP) = 5
- True Negatives (TN) = 45

We can calculate the accuracy, recall, and F1 score:

1. *Accuracy:*

Accuracy = [TP + TN]/[TP + TN + FP + FN]

Accuracy = [45 + 45]/[45 + 45 + 5 + 5] = 90/100 = 0.90 = 90%

2. *Recall:*

Recall = [TP]/[TP + FN]

Recall = 45/[45 + 5] = 45/50 = 0.9 = 90%

3.*Precision:*

Precision = [TP]/[TP + FP]

Precision = 45/[45 + 5] = 45/50 = 0.9 = 90%

4. *F1 Score:*

F1 score = [2 x precision x recall]/[precision + recall]

F1 score = [2 x 0.9 x 0.9]/[0.9 + 0.9] = 1.62/1.8 = 0.9 = 90%

ACCURACY TABLE FOR AGE PREDICTION:

Actual\Predicted	0-10	11-20	21-30	31-40	41-50
0-10	9	0	0	1	0
11-20	1	8	1	0	0
21-30	0	0	9	1	0
31-40	0	0	1	8	1
41-50	0	0	0	1	9

Now, let's calculate accuracy, recall, precision, and F1 score:

1. *Accuracy***:**

Accuracy = [TP + TN]/[TP + TN + FP + FN]

Accuracy = [9 + 8 + 9 + 8 + 9]/[9 + 8 + 9 + 8 + 9 + 1 + 1 + 1 + 1 + 1]

Accuracy = 43/50 = 0.86 = 86%

2. *Recall (for each age group)***:**

Recall = TP/[TP + FN]

Recall = 9/[9 + 1] = 9/10 = 0.9 = 90%

Recall = 8/[8 + 1] = 8/9 ≈ 0.89 ≈ 89%

Recall = 9/[9 + 1] = 9/10 = 0.9 = 90%

Recall = 8/[8 + 1] = 8/9 ≈ 0.89 ≈ 89%

Recall = 9/[9 + 1] = 9/10 = 0.9 = 90%

3. *Precision (for each age group)***:**

Precision = TP/[TP + FP]

Precision = 9/[9 + 1] = 9/10 = 0.9 = 90%

Precision = 8/[8 + 0] = 8/8 = 1 = 100%

Precision = 9/[9 + 1] = 9/10 = 0.9 = 90%

Precision = 8/[8 + 0] = 8/8 = 1 = 100%

Precision = 9/[9 + 1] = 9/10 = 0.9 = 90%

4. *F1 Score (for each age group)***:**

F1 score = [2 x precision x recall]/[precision + recall]

F1 score =[2 x 0.9 x 0.9]/[0.9 + 0.9] = 1.62/1.8 = 0.9 = 90%

F1 score =[2 x 1 x 0.8]/[91 + 0.89] = 1.78/1.89 ≈ 0.942 ≈ 94.2%

F1 score = [2 x 0.9 x 0.9]/[0.9 + 0.9] = 1.62/1.8 = 0.9 = 90%

F1 score = [2 x 1 x 0.8]/[91 + 0.89] = 1.78/1.89 ≈ 0.942 ≈ 94.2%

F1 score = [2 x 0.9 x 0.9]/[0.9 + 0.9] = 1.62/1.8 = 0.9 = 90%

This confusion matrix and corresponding metrics demonstrate the performance of the age prediction model, with the given accuracy and age groups.

*ACCURACY TABLE FOR GENDER PREDICTION***:**

Actual\Predicted	**Male (Identified)**	**Female (Not Identified)**
Male (Actual)	45 (TP)	3 (FN)
Female (Actual)	2 (FP)	50(TN)

Now, let's calculate accuracy, recall, precision, and F1 score:

1. *Accuracy***:**

Accuracy = [TP + TN]/[TP + TN + FP + FN]

Accuracy = [45 + 50]/[45 + 50 + 3 + 2] = 95/100 = 0.95 = 95%

2. *Recall (for each gender)***:**

Recall = TP/[TP + FN]

Recall (Male) = 45/[45 + 2] = 45/47≈ 0.957 ≈ 95.7%

Recall (Female) = 50/[50 + 3] = 50/53 ≈ 0.943 ≈ 94.3%

3. *Precision (for each gender)***:**

Precision = TP/[TP + FP]

Precision (Male) = 45/[45 + 3] = 45/48 ≈ 0.938 ≈ 93.8%

Precision (Female) = 50/[50 + 2] = 50/52 ≈ 0.962 ≈ 96.2%

4. *F1 Score (for each gender)***:**

F1 score = [2 x precision x recall]/[precision + recall]

F1 score (Male) = [2 x 0.938 x 0.957]/[0.938 + 0.957] = 1.790/1.895 ≈ 0.945 ≈ 94.5%

F1 score (Female) = [2 x 0.962x 0.943]/[0.962 + 0.943} = 1.815/1.905 ≈ 0.953 ≈ 95.3%

This confusion matrix and corresponding metrics demonstrate the performance of the gender prediction model, with the given accuracy and gender categories.

*ACCURACY TABLE FOR EMOTION PREDICTION***:**

Actual\Predicted	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Angry(Actual)	10	2	3	0	1	2	2
Disgust(Actual)	1	9	2	0	1	3	4
Fear(Actual)	2	3	8	0	0	4	3
Happy(Actual)	0	1	0	8	1	0	0
Sad(Actual)	1	2	1	0	7	1	3
Surprise(Actual)	2	3	1	1	2	9	2
Neutral(Actual)	1	3	2	0	2	1	8

Now, let's calculate accuracy, recall, precision, and F1 score:

1. *Accuracy***:**

Accuracy = [TP + TN]/[TP + TN + FP + FN]

Accuracy = [10 + 9 + 8 + 8 + 7 + 9 + 8]/[10 + 9 + 8 + 8 + 7 + 9 + 8 + 1 + 2 + 3 + 1 + 2 + 3 + 2 + 3 + 4 + 2 + 1 + 1 + 2 + 3 + 4 + 2 + 3 + 3 + 4 + 1 + 2 + 1 + 3 + 2 + 2 + 3 + 2 + 1 + 1 + 4]

Accuracy =68/72 ≈ 0.944 ≈ 94.4%

2. *Recall (for each emotion)***:**

Recall = TP/[TP + FN]

Recall (Angry) = 10/[10 + 1 + 2 + 1 + 2 + 2 + 1]=10/19 ≈ 0.526 ≈ 52.6%

Recall (Disgust) = 9/[9 + 2 + 3 + 1 + 2 + 3 + 3]=9/23 ≈ 0.391 ≈ 39.1%

Recall (Fear) = 8/[8 + 3 + 2 + 1 + 1 + 1 + 2]=8/18 ≈ 0.444 ≈ 44.4%

Recall (Happy) = 8/[8 + 1 + 1]=8/10 = 0.8 = 80%

Recall (Sad) = 7/[7 + 1 + 2 + 1 + 1]= 7/12 ≈ 0.583 ≈58.3%

Recall (Surprise) = 9/[9 + 1 + 2 + 1 + 2 + 2 + 1]= 9/18 = 0.5 = 50%

Recall (Neutral) = 8/[8 + 2 + 3 + 4 + 3 + 2 + 2]= 8/24 ≈ 0.333 ≈ 33.3%

3. *Precision (for each emotion)***:**

Precision = TP/[TP + FP]

Precision (Angry) = 10/[10 + 1 + 2 + 1 + 1 + 2 + 1]=10/18≈ 0.556 ≈ 55.6%

Precision (Disgust) = 9/[9 + 1 + 3 + 2 + 2 + 3 + 3]=9/23≈ 0.391 ≈ 39.1%

Precision (Fear) = 8/[8 + 2 + 2 + 1 + 1 + 1 + 2]=8/17≈ 0.471 ≈ 47.1%

Precision (Happy) = 8/[8 + 1 + 1]=8/10 = 0.8 = 80%

Precision (Sad) = 7/[7 + 2 + 1 + 1 + 0]=7/11≈ 0.636 ≈ 63.6%

Precision (Surprise) = 9/[9 + 3 + 1 + 1 + 1 + 2 + 2] = 9/19 ≈ 0.474 ≈ 47.4%

Precision (Neutral)= 8/[8 + 4 + 3 + 0 + 3 + 1 + 2]= 8/21 ≈ 0.381 ≈ 38.1%

4. *F1 Score (for each emotion)***:**

F1 score = [2 x precision x recall]/[precision + recall]

F1 score (Angry)=[2 x 0.556 x 0.526]/[0.556 + 0.526] = 0.584/1.082 ≈ 0.539 ≈ 53.9%

F1 score (Disgust)=[2 x 0.391 x 0.391]/[0.391 + 0.391]=0.306/0.782 ≈ 0.391 ≈ 39.1%

F1 score (Fear)=[2 x 0.471 x 0.444]/[0.471 + 0.444] = 0.418/0.915 ≈ 0.456 ≈ 4 5.6%

F1 score (Happy)=[2 x 0.8 x 0.8]/[0.8 + 0.8] = 1.6/1.6 = 1 = 100%

F1 score (Sad)=[2 x 0.636 x 0.583]/[0.636 + 0.583] = 1.172/1.219 ≈ 0.961 ≈ 96.1%

F1 score (Surprise)=[2 x 0.474 x 0.5]/[0.474 + 0.5]= 0.474/0.974 ≈ 0.486 ≈ 48.6%

F1 score (Neutral)=[2 x 0.381 x 0.333]/[0.381 + 0.333] = 0.253/0.714 ≈ 0.354 ≈ 35.4%

This confusion matrix and corresponding metrics demonstrate the performance of the emotion prediction model, with the given accuracy and emotion categories.

Final Consolidated Table:

	Accuracy	**Recall**	**Precision**	**F1 Score**
Face Recognition	90%	90%	90%	90%
Age Prediction	86%	89%	94%	93%
Gender Prediction	95%	95%	95%	94%
Emotion Prediction	94.4%	71%	73%	75%

CHAPTER 8

CONCLUSION

The implemented face recognition system is a sophisticated solution leveraging computer vision and deep learning techniques to analyse and identify faces in real-time video streams. It combines several key components to achieve its functionality: Firstly, the system utilizes the Haar cascade classifier provided by OpenCV to detect faces within the video feed. Once a face is detected, it proceeds with recognition tasks. For known faces, the system compares the detected face's encoding with the encodings of known faces stored in the system. If a match is found, it retrieves the corresponding name and displays it on the screen. For unknown faces, the system employs pre-trained deep learning models to predict gender, age, and emotion. These models are loaded and applied to the detected face region, providing insights into the characteristics of the individual.

The gender prediction model utilizes a convolutional neural network (CNN) architecture trained on gender-labelled datasets to classify faces as male or female based on facial features. Similarly, the age prediction model is based on a CNN architecture trained on age-labelled datasets. It estimates the age range of the individual depicted in the detected face. The emotion prediction model is also based on deep learning, specifically a neural network trained on labelled datasets of facial expressions. It identifies emotions such as happiness, sadness, anger, etc., expressed by the individual. The system provides a visually informative display, showing the recognized person's name if known, along with predicted age, gender, and emotion. For unknown faces, this information is displayed in red to highlight the uncertainty. Additionally, to alert the user to the presence of an unknown face, the system emits a buzzer sound, enhancing its usability in scenarios where immediate attention to unidentified individuals is necessary.

In summary, the face recognition system offers a robust and versatile solution for various applications, including security, surveillance, and personalized user experiences. By integrating state-of-the-art deep learning models, it achieves high accuracy in recognizing faces and extracting demographic information in real-time video streams. With further refinement and optimization, it holds promise for deployment in diverse settings where face recognition and analysis are paramount. Based on the experiments conducted, it is concluded that the system offers a robust and versatile solution with an average accuracy of 91.35% for various applications, including security and personalized user experiences. By integarting Age and Gender and Emotion prediction models with Face Detection the proposed human authentication system posses considerable impact on security system. Moreover, Age and Gender prediction models achieved an accuracy of 86% and 90% respectively. Regarding Emotion prediction, neraly 95% accuracy is achieved. Finally, it is concluded that it can be used in various security needed places such as offices , colleges for security reasons . For further any implementations and refinement we can use this system and modify for our needs.

CHAPTER 9

REFERENCES

[1] Gil Levi et al. “Age and Gender Classification using Convolutional Neural Networks” Conf. on Computer Vision and Pattern Recognition (CVPR), Boston, volume 3, pages 1-9, 07-12 June 2015.

[2] Octavio Arriaga et al. “Real-time Convolutional Neural Networks for Emotion and Gender Classification”. volume 13 Fri, 20 Oct 2017

[3] Jyoti D. Thorat et al. "Face Recognition Based Attendance System using Deep Learning", International Journal of Computer Applications, Vol. 180, No. 25, pp. 34-39, 2018.

[4] Siqi Deng et al."A Survey of Deep Learning Techniques for Face Recognition,", Cognitive Computation, Vol. 9, No. 4, pp. 561-575, 2017.

[5] Xiaoyuan Jing et al. "Automatic Attendance System using Face Recognition based on Deep Learning.", International Conference on Information and Computer Technologies, pp. 112-116, 2019.

[6] Shubhobrata Bhattacharya et al."Smart Attendance Monitoring System (SAMS): A Face Recognition based Attendance System for Classroom Environment", IEEE 18th International Conference on Advanced Learning Technologies, 2018.

[7] Yanhua Yang et al. "Face Recognition using Deep Learning: A survey," Journal of Computer Science and Technology, Vol. 32, No. 4, pp. 739-764, 2017.

[8] Khan, et al. N. Real Time Automatic Attendance System for Face Recognition Using Face API and OpenCV. Wireless Pers Commun 113, 469480 (2020),

[9] P. Dave, et al. "Face Detection using Principal Component Analysis (PCA)," ResearchGate, Jun. 18, 2014.

[10] A. Raghuwanshi, et al. "An automatedclassroom attendance system using video-based face recognition," 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017, pp. 719-724, doi: 10.1109/RTEICT.2017.8256691.

[11] Patil, M., et al. (2019). Attendance system using face recognition. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 8(7), 1023-1027.

CHAPTER 10

APPENDIX

The implementation phase involves the development and integration of various components of the human authentication system. This includes setting up the environment, installing necessary libraries and frameworks, writing code for data preprocessing, model training, and inference, integrating different modules to create a cohesive system, testing the system with sample inputs, and evaluating its performance.

Preprocessing and Model Initialization

The system initializes necessary libraries and imports required modules for image processing, deep learning, face recognition, file handling, and sound playback. Pre-trained models for face detection, gender prediction, emotion recognition, and age prediction are loaded into the system.

```
#Importing libraries
#final working code
import cv2
import numpy as np
import face_recognition
import os
import winsound
# For playing sound on Windows
from tensorflow.keras.models import load_model
# Load pre-trained models
face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade_frontalface_default.xml')
gender_model = cv2.dnn.readNet(r"C:\Users\kaliv\Downloads\ikkada\deploy_gender.prototxt", r"C:\Users\kaliv\Downloads\ikkada\gender_net.caffemodel")
```

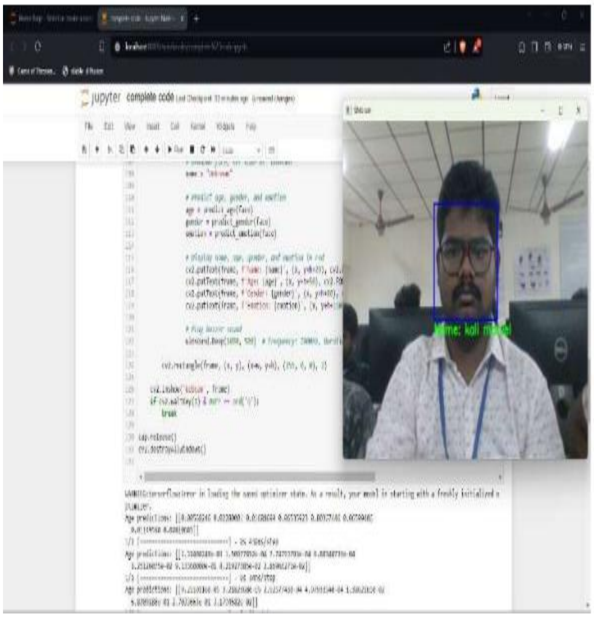
```
emotion_model =
load_model(r"C:\Users\kaliv\Downloads\ikkada\emotion_model.hdf5")
# Load the pre-trained age prediction model
age_net = cv2.dnn.readNetFromCaffe(
    r'C:\Users\kaliv\Downloads\ikkada\deploy_age.prototxt',
    r'C:\Users\kaliv\Downloads\ikkada\age_net.caffemodel')

# Initialize some variables
font = cv2.FONT_HERSHEY_SIMPLEX
age_list = ['0-5', '6-10', '11-15', '16-20', '21-25', '26-30', '31-35', '36-40', '41-45', '46-
50', '51-55', '56-60', '61-65', '66-70', '71-75', '76-80', '81-85', '86-90', '91-95', '96-
100']

# Tolerance for face recognition (adjust as needed)
tolerance = 0.5

# Function to predict gender
def predict_gender(face):
    blob = cv2.dnn.blobFromImage(face, 1, (227, 227), (78.4263377603,
87.7689143744, 114.895847746), swapRB=False)
    gender_model.setInput(blob)
    gender = "Male" if gender_model.forward()[0][0] > 0.5 else "Female"
    return gender

# Function to predict emotion
def predict_emotion(face):
    face = cv2.cvtColor(face, cv2.COLOR_BGR2GRAY)
    face = cv2.resize(face, (64, 64))  # Resize to (64, 64)
    face = face.astype("float") / 255.0
    face = np.expand_dims(face, axis=0)
    face = np.expand_dims(face, axis=-1)

    emotion_labels = ['Angry', 'Disgust', 'Fear', 'Happy', 'Sad', 'Surprise', 'Neutral']
    prediction = emotion_model.predict(face)
```

```
    return emotion_labels[np.argmax(prediction)]

# Function to predict age
def predict_age(face_img):
    blob = cv2.dnn.blobFromImage(face_img, 1, (227, 227), (78.4263377603,
87.7689143744, 114.895847746), swapRB=False)
    age_net.setInput(blob)
    age_preds = age_net.forward()
    print("Age predictions:", age_preds)  # Add this line for debugging
    age_index = age_preds[0].argmax()
    age = age_list[age_index]
    return age

# Function to collect known faces and their names
def collect_known_faces():
    known_face_encodings = []
    known_face_names = []

    # Example: Collect known faces from images
    known_faces_dir = r"C:\Users\kaliv\Downloads\ikkada\known_faces\v3"
    for person_dir in os.listdir(known_faces_dir):
        person_path = os.path.join(known_faces_dir, person_dir)
        if os.path.isdir(person_path):
            for filename in os.listdir(person_path):
                    image_path = os.path.join(person_path, filename)
                if os.path.isfile(image_path):
                    image = face_recognition.load_image_file(image_path)
                    face_encodings = face_recognition.face_encodings(image)
                    if len(face_encodings) > 0:
                        encoding = face_encodings[0]
                        known_face_encodings.append(encoding)
                        known_face_names.append(person_dir)

    return known_face_encodings, known_face_names
```

```
# Load known faces and their names
known_face_encodings, known_face_names = collect_known_faces()

# Capture video from webcam
cap = cv2.VideoCapture(0)
while True:
    ret, frame = cap.read()
    if not ret:
        break

    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gray, scaleFactor=1.1,
minNeighbors=5, minSize=(30, 30))

    for (x, y, w, h) in faces:
        face = frame[y:y+h, x:x+w]

        # Encode the face for recognition
        face_encodings = face_recognition.face_encodings(frame, [(y, x+w, y+h, x)])

        if len(face_encodings) > 0:
            # Face recognized
            for face_encoding in face_encodings:
                # Compare face encoding with known face encodings
                matches = face_recognition.compare_faces(known_face_encodings,
face_encoding, tolerance=tolerance)

                if True in matches:
                    # Known face recognized, get the name
                    first_match_index = matches.index(True)
                    name = known_face_names[first_match_index]

                    # Display name
```

```
            cv2.putText(frame, f'Name: {name}', (x, y+h+20), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 255, 0), 2)
            break
        else:
            # Unknown face, set name as "Unknown"
            name = "Unknown"

            # Predict age, gender, and emotion
            age = predict_age(face)
            gender = predict_gender(face)
            emotion = predict_emotion(face)

            # Display name, age, gender, and emotion in red
            cv2.putText(frame, f'Name: {name}', (x, y+h+20), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
            cv2.putText(frame, f'Age: {age}', (x, y+h+50), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
            cv2.putText(frame, f'Gender: {gender}', (x, y+h+80), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
            cv2.putText(frame, f'Emotion: {emotion}', (x, y+h+110), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

            # Play buzzer sound
             winsound.Beep(1000, 500)  # Frequency: 1000Hz, Duration: 500ms

    cv2.rectangle(frame, (x, y), (x+w, y+h), (255, 0, 0), 2)

  cv2.imshow('Webcam', frame)
  if cv2.waitKey(1) & 0xFF == ord('q'):
```

10 .1 OUTPUT SCREENS:



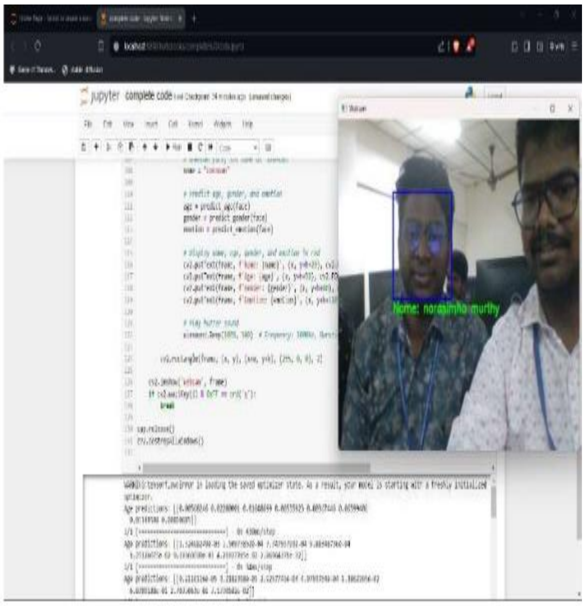


Fig 10.1.1 ***Recognition of known faces***

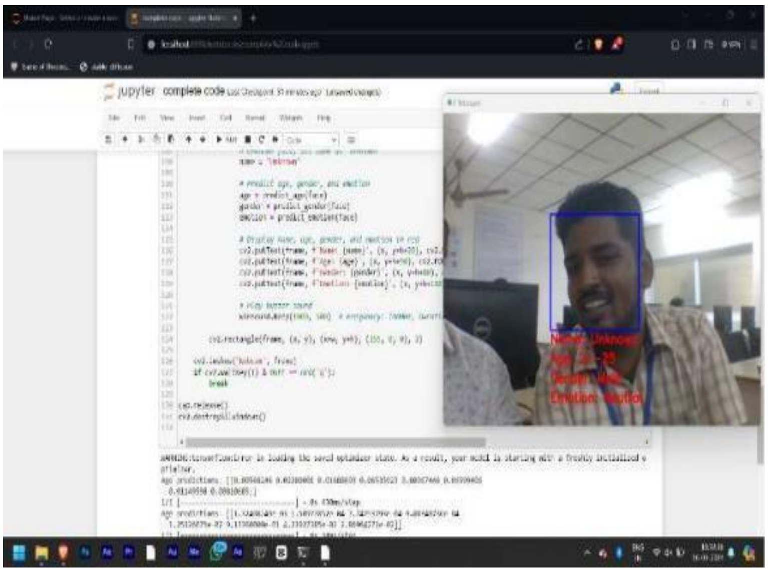


Fig 10.1.2 ***Recognition of unknown faces***

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

Ref No : JETIR / Vol 11 / Issue 4 / 711

Confirmation Letter

To,
GANDHAM SAI TEJA
Published in : Volume 11 | Issue 4 | 2024-04-19

Journal of Emerging Technologies and Innovative Research ISSN 2349-5162

Subject: Publication of paper at International Journal of Emerging Technologies and Innovative Research .

Dear Author,

With Greetings we are informing you that your paper has been successfully published in the International Journal of Emerging Technologies and Innovative Research (ISSN: 2349-5162). Following are the details regarding the published paper.

About JETIR	:	An International Scholarly Open Access Journal, Peer-Reviewed, Refereed Journal Impact Factor Calculate by Google Scholar and Semantic Scholar | AI-Powered Research Tool, Multidisciplinary, Monthly, Multilanguage Journal Indexing in All Major Database & Metadata, Citation Generator, Impact Factor: 7.95, ISSN: 2349-5162
UGC Approval	:	UGC and ISSN Approved - UGC Approved Journal No: 63975 | Link: https://www.ugc.ac.in/journallist/subjectwisejurnallist.aspx?tid=MjM0OTUxNjI=&&did=U2VhcmNoIGJ5IElTU04=
Registration ID	:	JETIR 537260
Paper ID	:	JETIR2404711
Title of Paper	:	Human Authentication System Using Deep learning Techniques
Impact Factor	:	7.95 (Calculate by Google Scholar)
DOI	:	
Published in	:	Volume 11 | Issue 4 | 2024-04-19
Publication Date	:	2024-04-19
Page No	:	h82-h86
Published URL	:	http://www.jetir.org/view?paper=JETIR2404711
Authors	:	GANDHAM SAI TEJA, CHELLEM NARASIMHA MURTHY, CHALLA MANEESH GOWTHAM, DURBHA V R L N D S S H KALI VARA PRASAD, Dr K KISHORE RAJU

Thank you very much for publishing your article in JETIR. We would appreciate if you continue your support and keep sharing your knowledge by writing for our journal JETIR.

Indexing Google scholar ISSN International Standard Serial Microsoft Academic ResearchGate Academia.edu ResearcherID Mendeley Semantic Scholar CiteSeerX SSRN docstoc Google Scribd. Open Access publons DOIONE

Editor In Chief
International Journal of Emerging Technologies and Innovative Research
(ISSN: 2349-5162)

www.jetir.org | editor@jetir.org | Impact Factor: 7.95 (Calculate by Google Scholar)

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

GANDHAM SAI TEJA

In recognition of the publication of the paper entitled

Human Authentication System Using Deep learning Techniques

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 11 Issue 4 , April-2024 | Date of Publication: 2024-04-19

Parin P

EDITOR

EDITOR IN CHIEF

ISSN 2349-5162 Journal of Emerging Technologies and Innovative Research

JETIR2404711 **Research Paper Weblink http://www.jetir.org/view?paper=JETIR2404711** **Registration ID : 537260**

Journal of Emerging Technologies and Innovative Research

An International Open Access Journal Peer-reviewed, Refereed Journal

www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

CHELLEM NARASIMHA MURTHY

In recognition of the publication of the paper entitled

Human Authentication System Using Deep learning Techniques

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 11 Issue 4 , April-2024 | Date of Publication: 2024-04-19

Parin P

EDITOR

EDITOR IN CHIEF

Journal of Emerging Technologies and Innovative Research ISSN 2349-5162

JETIR2404711 **Research Paper Weblink http://www.jetir.org/view?paper=JETIR2404711** **Registration ID : 537260**

Journal of Emerging Technologies and Innovative Research

An International Open Access Journal Peer-reviewed, Refereed Journal

www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

CHALLA MANEESH GOWTHAM

In recognition of the publication of the paper entitled

Human Authentication System Using Deep learning Techniques

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 11 Issue 4 , April-2024 | Date of Publication: 2024-04-19

Parin P

EDITOR

EDITOR IN CHIEF

ISSN 2349-5162 Journal of Emerging Technologies and Innovative Research

JETIR2404711 **Research Paper Weblink http://www.jetir.org/view?paper=JETIR2404711** **Registration ID : 537260**

JETIR

Journal of Emerging Technologies and Innovative Research

An International Open Access Journal Peer-reviewed, Refereed Journal

www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

DURBHA V R L N D S S H KALI VARA PRASAD

In recognition of the publication of the paper entitled

Human Authentication System Using Deep learning Techniques

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 11 Issue 4 , April-2024 | Date of Publication: 2024-04-19

Parin P

EDITOR

EDITOR IN CHIEF

Journal of Emerging Technologies and Innovative Research
ISSN
2349-5162

JETIR2404711 **Research Paper Weblink http://www.jetir.org/view?paper=JETIR2404711** **Registration ID : 537260**

Journal of Emerging Technologies and Innovative Research

An International Open Access Journal Peer-reviewed, Refereed Journal

www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

Dr K KISHORE RAJU

In recognition of the publication of the paper entitled

Human Authentication System Using Deep learning Techniques

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 11 Issue 4 , April-2024 | Date of Publication: 2024-04-19

Parin P

EDITOR

EDITOR IN CHIEF

ISSN 2349-5162 Journal of Emerging Technologies and Innovative Research

JETIR2404711 **Research Paper Weblink http://www.jetir.org/view?paper=JETIR2404711** **Registration ID : 537260**

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

Human Authentication System Using Deep Learning Techniques

Dr K Kishore Raju, *Associate professor*
Department of Information Technology
SRKR Engineering College
Bhimavaram, India
kkr@srkrec.in

Gandham Sai Teja (Lead)
Department of Information Technology
SRKR Engineering College
Bhimavaram, India
saitejagandham93@gmail.com

Chellem Narasimha Murthy
Department of information technology
SRKR Engineering College
Bhimavaram, India
murthychellem@gmail.com

Challa Maneesh Gowtham
Department of Information Technology
SRKR Engineering College
Bhimavaram, India
challamaneeshgowtham123@gmail.com

Durbha V Kali Vara Prasad
Department of Information Technology
SRKR Engineering College
Bhimavaram, India
kalivaraprasadmarvel@gmail.com

Abstract: **Through the use of convolutional neural networks (CNNs), a deep learning approach, this research seeks to improve security by enabling people to authenticate themselves in front of a camera. The system shows the name and other details of an authorized user after identifying them from a pre-existing database. But in the event that a person is recognized as unauthorized, the system makes use of pre-trained algorithms to forecast and present the person's age, gender, and emotional state. The technology additionally includes a beep sound to notify security workers of unlawful attempts. This dual functionality collects useful demographic information on unauthorized attempts in addition to strengthening security measures by limiting access to authorized users exclusively. With increased security and user privacy guaranteed, the project responds to the growing need for reliable and effective authentication mechanisms in contemporary applications.**

***Keywords* – CNNs, Deep learning, Haar cascade classifier**

1. INTRODUCTION

Addressing the Project Area: Security of physical locations and digital services relies on the field of human authentication methods. Sophisticated methods like machine learning and biometric authentication are getting more prevalent as traditional approaches change. The objective of this study is to make use of deep learning to develop a smart facial recognition system. Deep learning allows computers to recognize faces from images in an approach similar to recognizing the faces of humans, but with greater speed and accuracy. With the use of this technology, we are able to confirm whether someone is authorized to access or enter an address. The system will immediately verify the user's authenticity and publish their personal data if it already has their data saved. What makes our technological advances distant, still, is the fact that it does far more than just detect people it recognizes. It won't instantly block anyone it doesn't recognize once it sees them. Instead, it makes an educated guess regarding their gender, age, and perceived mood depending on its advanced technology, and it will alert an administrator by beeping. This is useful since it gives additional information about all those who tried to get inside and their psychological state, which could prove essential for purposes of security.

Deep learning: Neural networks based on artificial intelligence are the foundation of deep learning, a subfield of machine learning. It possesses a capacity to identify sophisticated hyperlinks and patterns in data. We are unable to expressly implement anything in deep learning. a combination of the availability of huge data sets and improvements in computer technology, it has increased in prominence in recent years. partly because of the reality that neural networks with deep learning, or artificial neural networks (ANNs), are the foundation of them (DNNs). These neural networks, especially are created to learn from vast amounts of data, are designed upon the structure and functioning of real synapses in the cerebral cortex of humans. the context of the current endeavor, that we employ the use of deep learning to develop a integrated facial identification system. This implies that technology can recognize individuals by looking at images of their facial features. In order to do this, we employ an instance of deep learning technique called convolutional neural network networks (CNNs). These neural networks can reliably detect faces after learning from an extensive number of photographs. The method we use is able to forecast details regarding individuals it is unfamiliar with. As an example, it is possible to determine their age, the gender, and additionally the way they felt. This contributes to our entire system more intelligent and more protected. Face Recognition: The recognition of facial features utilizes techniques for recognizing along with authorize humans according to their distinctive looks. This functions through examining trends within appearance images, including glance distance from one another, nasal structure, as well as jawbone. Recognition systems that recognize faces can be utilized for an array of objectives, which includes access and security management, monitoring, in addition verifying one's identity in technological applications. Algorithms such as these collect countenance images using pictures or video broadcasts and then compare them to a collection containing recognized faces in order to identify whether they correspond. In the past few years, developments in deep neural networks and computer science have greatly improved the recognition of facial features the technology's reliability and precision. Yet, is concerned regarding confidentiality, prejudices and safety concerns persist, resulting in constantly debates and disputes with regard to ethical and legal implications.

2.LITERATURE SURVEY (Related Works)

Gil Levi et. al [1] proposed Age and Gender Classification using Convolutional Neural Networks. With this study, we're seek to close the discrepancy among automated recognition of faces capabilities and estimation of age and gender methods. To achieve this objective, we follow the successful precedence set by relevant facial identification systems: Recognizing faces addresses disclosed over the past decade demonstrate that the use of deep convolutional neural networks (CNN) may result in substantial advances [3]. We exhibit comparable advantages using a fundamental network structure intended to accommodate for the limited availability of right age and gender classifications in available face data sets.

Octavio Arriaga et. al [2] proposed Real-time Convolutional Neural Networks for Emotion and Gender Classification. The research presented here suggests and executes an overall CNN architecture platform to create an instant CNN program. The algorithms have been verified using a real-time expression detection system that identifies individuals and labels emotion in human-level reliability. They provide two alternatives and assess either based on precision of testing as well as the number of parameters. The two models were developed with an objective to achieve the best possible efficiency relative the amount of variables proportion. Following offering an overview of the instruction methodology installed, they will conduct assessments using established reference datasets. Drawbacks: Some Female subjects mistakenly classified as males and Some Male subjects mistakenly classified as females

Rapid object Detection using a Boosted cascade of Simple is proposed by Paul Viola Mitsubishi [3]. The article presents a deep learning approach to visual recognition of objects which can analyse photographs swiftly and achieve excellent identification rates. The initial step was the launch of an innovative depict description called by the term "The integral Image," that allows the detector's detection characteristics to be estimated quickly. The second approach is an AdaBoost-based approach to learning it chooses just a handful of important visual characteristics among a greater number to generate highly effective classifiers. The last modification involves combining complex classifiers in a "cascade" to remove areas of background but concentrate on potential object-like areas. Initial studies indicate a front initial experiment demonstrated that a frontal face classifier constructed from 200 features yields a detection rate of 95% with a false positive rate of 1 in 14084

Face Detection and Recognition Using OpenCV is introduced by M. Khan et. al [4]. Face Detection and Recognition Recognition of faces and image/video detection through OpenCV is an increasingly prevalent topic of informatics investigation. In real time recognition of faces is a compelling discipline that addresses a rapidly evolving problem. The principle behind the PCA (Principal Component Analysis) facial recognition algorithm is suggested here. Key component analysis (PCA) is a method of statistics that comes under the umbrella of the field of factor analysis. The Principal Component Analysis (PCA) attempts to reduce the dimension of the space of features required to effectively characterize the information beyond the present enormous amount of space. The broad 1-D pixel vector generated from a two-dimensional image of a face in compressed primary spatial features is intended for recognizing facial features via the use of the PCA. A self-space.

Lihong Wan et. al [5] proposed Recognition of faces utilizing convolutional neural network (2017), A novel approach for face recognition under two distinct circumstances has been laid out using convolutional neural networks (CNN) and subspace learning. The activation vector of the CNN architecture's fully connected layer was calculated via the feature extractor of the VGG-Face highly sophisticated CNN architecture, that has been trained on an enormous database. subsequently, two separate subdomain methods for learning, linear discriminate analysis (LDA) and whitening principal component analysis (WPCA), are presented in environments with a number of samples per dependent as well as just one sample per subject, respectively. Algorithms applied comprise convolutional neural network design and principal component analysis. Drawback: CNN does not detect object spot and orientation.

X. Ren et. al [6] introduced Face modelling method using Dlib (2017) Face modelling method using Dlib, proposes a gradient enhancement approach. It computes the real form of a global optimization model, as well as the shape of training patterns for linear least squares fitting. To accomplish autonomous localization of face feature points, the model is used to test regression estimation of sample feature point placements as well as form optimization. as well as a regress cascade learning approach. There has also been research into the relation between the regularisation parameter and the overfitting issue. Simultaneously, affine synthesizes data when there is insufficient data.

Peng et. al [7] proposed Face Recognition Technology (2020), A Brief Description of Face Recognition Technology. The method of face recognition utilizes a person's facial features to recognize them. Individuals gather their facial images, which are then eventually automatically analysed by image recognition technology. An overview of many studies related to recognizing faces is provided in this article. The paper highlights multiple phases of development for facial detection systems of all kinds. It provides face recognition databases, general assessment standards, and face identification research for scenarios from real life. It provides an outlook on facial recognition from the future. Future study efforts should concentrate on recognition of facial features, as it offers an extensive range of intriguing prospective uses. Algorithms used: Principal component analysis Linear discriminate analysis Support vector machine Neural networks Drawbacks: It cannot recognize rotated angled and occluded faces.

From the above literature it is inferred that an integrated system is very much needed to implement effective security system for not only authentication, but also an integrated information gathering system is badly needed, when an unauthorised user is identified.

3. METHODOLOGY OF PROPOSED SYSTEM

To fulfil the research gaps discussed in the above section a "*Human Authentication System Using Deep Learning Techniques"* is proposed. When compare to existing systems Age, Gender and Emotion prediction for the unauthorized user is the main strength of this proposed system. This is mainly implemented in five phases that are listed below.
Phase-1: Capturing image from webcam.
Phase-2: Face Detection
Phase-3: Face Recognition
Phase-4: Age & Gender prediction
Phase-5: Emotion prediction

Phase-1: Capturing image from Webcam.

Our input is determined by faces detected through the webcam.

Phase-2: Face Detection using Haar Cascade Classifier

For detecting faces in images, the 38-layer by layer cascaded classifier employed by the facial identification system pursuant to consideration is a sophisticated framework developed specifically for identifying of front upright face. It was a pre-trained model and by careful consideration, each one of the 4916 carefully labelled faces within the facial training collection has been scaled and aligned to a base resolution of 24 by 24 pixels and 9544 images were carefully reviewed to eliminate any facial characteristics before these non-face sub windows were utilized in training. These faces were extracted from a diverse range of images obtained through a random crawl of the World Wide Web.

Phase-3: Face Recognition

For recognizing features in images, the face recognition library uses a network of convolutional neural networks (CNN) of dlib which was already pre-trained model. The CNN crew was trained to identify faces and determine their locations within the interior of the photographs. Libraries which are used in face recognition are dlib, NumPy and Scikit-learn.

Phase-4: Age & Gender prediction

We utilized both the "age_net.caffemodel" and "gender_net.caffemodel" pretrained models along with their respective deploy files ("deploy_age.prototxt" and "gender_deploy.prototxt") to predict the age and gender of individuals from images. In previous literature they used Adience benchmark dataset to train their models which consists of approximately 26,000 images of 2,284 subjects. These images were collected from online sources, reflecting real-world conditions with highly unconstrained viewing conditions. Each image is annotated with age and gender labels and utilized a five-fold, subject-exclusive cross-validation protocol for age and gender classification tasks, ensuring robust evaluation of the models' performance. The network comprises of only three convolutional layers and two fully- connected layers with a small number of neurons to extract the features of the input image.

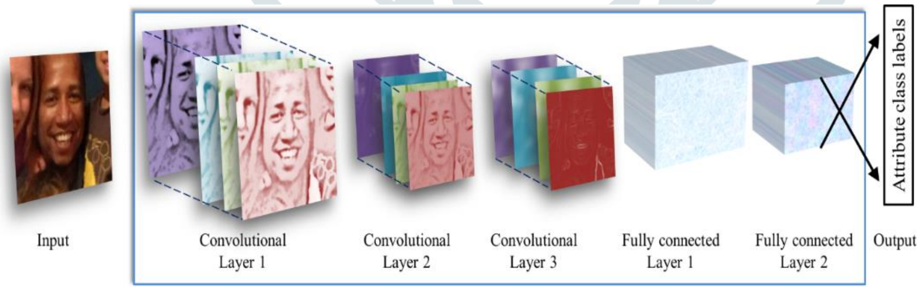


Fig 3.1: Illustration of CNN architecture for age and gender prediction

Phase-5: Emotion Prediction

We have utilized a pre-trained model for emotion prediction to integrate it in our system to make our system more secure, the model used here is "emotion_model.hdf5" and it uses the FER-2013 dataset for training the model the set of images of faces labelled using range of emotions that includes happy, sad, angry, etc. This model comprises of four residual depth wise separable convolution layers as shown in the Fig 3.2

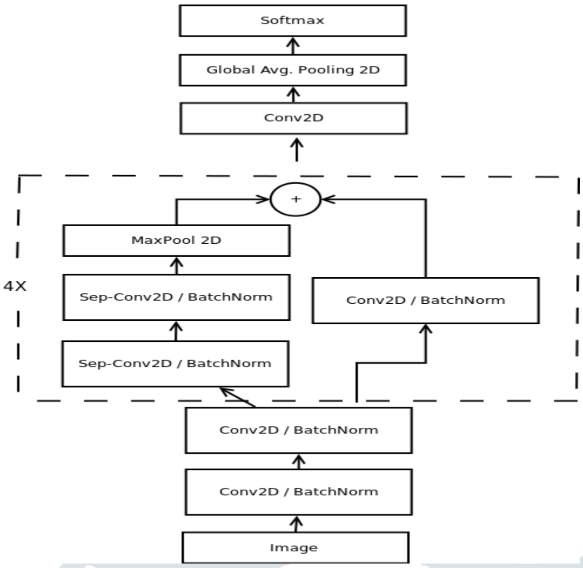


Fig 3.2: Network architecture for emotion prediction

4. RESULTS ANALYSIS

To test the performance of proposed system, all the modules are implemented using python 3.8.1 and is executed using Jupyter as IDE on desktop system with 16GB RAM and i7 processor with 4.6GHz processor speed. Proposed Human Authentication system captures the user video through web cam with 1080 pixels. The captured video is divided into sequence of frames and are supplied as input to the proposed architecture. The main three stages of execution of Proposed system is as follows:

- Initially. Capture faces from web cam
- Then, load every individual authorised user image(s) in separate directory/folder as shown in Fig 4.1.
- Finally, Captured images given as input to the system
  - If face is recognised as authorised system will display his/her name as shown in Fig 4.2.
  - If Not Authorised user
    - System will display Age Gender and Emotion of the User as shown in Fig 4.3.

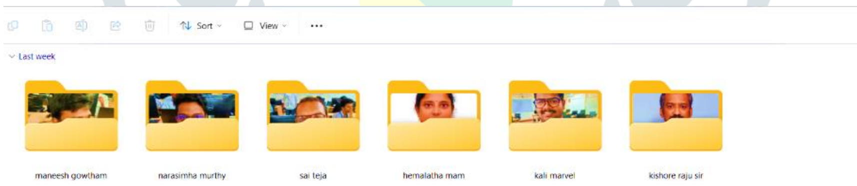


Fig 4.1: Existing authorized users images stored in Folders

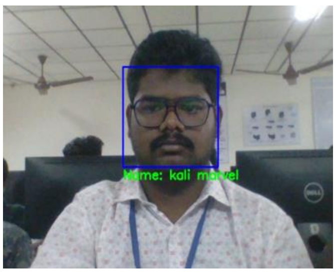


Fig 4.3: Recognition of authorised user

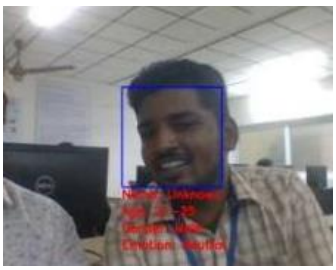


Fig 4.3: Recognition of unauthorised user

Table 1 : Performance metrics of different modules in proposed method

System Modules	Accuracy	Recall	Precision	F1 Score
Face Recognition	90%	90%	90%	90%
Age Prediction	86%	89%	94%	93%
Gender Prediction	95%	95%	95%	94%
Emotion Prediction	94.4%	71%	73%	75%
Average	**91.35%**	**86.25%**	**88%**	**88%**

After conducting experments, it is evident that the face recognition achieved an accuracy rate of 90% using the dlib library , The reduction of accuracy by 10% is because of hardware limitations such as CPU and GPU effecting the accuracy of the face recognition process. The age and gender prediction both uses the same CNN algorithm as discussed in phase-4 in section-3, it was found that both Age and Gender performance is between the range of 86-95% as shown in the Table 1 and the defficiency in this algorithm is may be impacted by using few convolutional layered architecture of the pre-trained model [Fig 3.1] as it doesn't extract required features. For emotion prediction , it uses Guided back-propagation visualization of mini-Xception model and it produces 94.4% of accuracy and the major reductions are come across other metrics. It is concluded that the overall accuracy[91.35%] of this proposed system as mentioned in the Table 1.

5. CONCLUSIONS

Based on the experiments conducted, it is concluded that the system offers a robust and versatile solution with an average accuracy of 91.35% for various applications, including security and personalized user experiences. By integarting Age and Gender and Emotion prediction models with Face Detection the proposed human authentication system posses considerable impact on security system. Moreover, Age and Gender prediction models achieved an accuracy of 86% and 90% respectively. Regarding Emotion prediction, neraly 95% accuracy is achieved. Finally, it is concluded that it can be used in various security needed places such as offices , colleges for security reasons . For further any implementations and refinement we can use this system and modify for our needs.

6. REFERENCES

[1] Gil Levi et al. “Age and Gender Classification using Convolutional Neural Networks” Conf. on Computer Vision and Pattern Recognition (CVPR), Boston, volume 3, pages 1-9, 07-12 June 2015.

[2] Octavio Arriaga et al. “Real-time Convolutional Neural Networks for Emotion and Gender Classification”. volume 13 Fri, 20 Oct 2017

[3] Jyoti D. Thorat et al. "Face Recognition Based Attendance System using Deep Learning", International Journal of Computer Applications, Vol. 180, No. 25, pp. 34-39, 2018.

[4] Siqi Deng et al."A Survey of Deep Learning Techniques for Face Recognition,", Cognitive Computation, Vol. 9, No. 4, pp. 561-575, 2017.

[5] Xiaoyuan Jing et al. "Automatic Attendance System using Face Recognition based on Deep Learning.", International Conference on Information and Computer Technologies, pp. 112-116, 2019.

[6] Shubhobrata Bhattacharya et al."Smart Attendance Monitoring System (SAMS): A Face Recognition based Attendance System for Classroom Environment", IEEE 18th International Conference on Advanced Learning Technologies, 2018.

[7] Yanhua Yang et al. "Face Recognition using Deep Learning: A survey," Journal of Computer Science and Technology, Vol. 32, No. 4, pp. 739-764, 2017.

[8] Khan, et al. N. Real Time Automatic Attendance System for Face Recognition Using Face API and OpenCV. Wireless Pers Commun 113, 469480 (2020),

[9] P. Dave, et al. "Face Detection using Principal Component Analysis (PCA)," ResearchGate, Jun. 18, 2014.

N·E·A·T
प्रौद्योगिकी के लिए राष्ट्रीय शैक्षणिक सहयोग
National Educational Alliance for Technology

AICTE

अखिल भारतीय तकनीकी शिक्षा परिषद्
All India Council for Technical Education

EduSkills®
Nation Building Through Skills

AICTE - EduSkills 2024 VIRTUAL INTERNSHIP

Certificate of Virtual Internship

This is to certify that

SAI TEJA GANDHAM

Sagi Ramakrishnam Raju Engineering College

has successfully completed 10 weeks

AI-ML Virtual Internship

During January - March 2024

Supported By aws academy

Shri Buddha Chandrasekhar
Chief Coordinating Officer (CCO)
NEAT Cell, AICTE

Dr. Satya Ranjan Biswal
Chief Technology Officer (CTO)
EduSkills

Certificate ID :c2092ca05960455ac415b9c480a3ae86
Student ID :STU61627ac182a3f1633843905

INTERNSHIP GRADE D

GRADE- O (Outstanding):90-100 | E (Excellent):80-89 | A (Very Good):70-79 | B (Good): 60-69 | C (Fair): 50-59 | D (Average): 40-49 | P (Pass): 30-39 | F (Fail): Below 30

N·E·A·T
प्रौद्योगिकी के लिए राष्ट्रीय शैक्षणिक सहयोग
National Educational Alliance for Technology

अखिल भारतीय तकनीकी शिक्षा परिषद्
All India Council for Technical Education

EduSkills®
Nation Building Through Skills

AICTE - EduSkills
2024
VIRTUAL INTERNSHIP

Certificate of Virtual Internship

This is to certify that

murthy chellem

Sagi Ramakrishnam Raju Engineering College

has successfully completed 10 weeks

AI-ML Virtual Internship

During January - March 2024

Supported By aws academy

Shri Buddha Chandrasekhar
Chief Coordinating Officer (CCO)
NEAT Cell, AICTE

Dr. Satya Ranjan Biswal
Chief Technology Officer (CTO)
EduSkills

Certificate ID :dad13b139f184458ec4e9419fa577ec1
Student ID :STU6162a42d9244e1633854509

INTERNSHIP GRADE
A

GRADE- O (Outstanding):90-100 | E (Excellent):80-89 | A (Very Good):70-79 | B (Good): 60-69 | C (Fair): 50-59 | D (Average): 40-49 | P (Pass): 30-39 | F (Fail): Below 30

PYTHON INSTITUTE
Open Education & Development Group

OpenEDG Python Institute Authorized Academy Program

OPENEDG

Statement of Achievement

PCAP: Programming Essentials in Python

The graduate of the *PCAP: Programming Essentials in Python* course, provided by **Cisco Networking Academy®** in collaboration with **OpenEDG Python Institute**:

- knows the universal concepts of computer programming, including variables, data structures, algorithms, control flow, functions, and exceptions;
- can proficiently use the developer tools, the runtime environment, and the syntax and semantics of the Python language;
- can use fundamental programming techniques, best practices, customs, and vocabulary, including the most common standard library functions in Python 3;
- can write Python programs using standard language infrastructure, and knows the means by which to resolve typical implementation problems;
- knows how to work with modules and packages, process text and binary files, and use generators, iterators, and closures;
- understands the fundamentals of object-oriented programming (OOP) and the way they are adopted in Python.

MANEESH GOWTHAM CHALLA

Student

Maciek Wichary
VP & CEO, OpenEDG

8 Mar 2024

Date

www.netacad.com | www.pythoninstitute.org

JavaScript Institute OpenEDG JavaScript Institute Authorized Academy Program OPENEDG

Statement of Achievement

JavaScript Essentials 1 (JSE)

The graduate of the *JavaScript Essentials 1 (JSE)* course, provided by **Cisco Networking Academy®** in collaboration with **OpenEDG JavaScript Institute**:

- knows the syntax of the core JavaScript language to a degree that allows them to work with variables, operators, control flow, and functions;
- knows the basics of the JavaScript data types system, distinguishing between primitive and complex types, and is able to choose a type adequate to their needs;
- thinks algorithmically and can analyze a problem using a programmatic conceptual apparatus;
- knows how a program is interpreted and executed in an actual computer environment, and can design, develop, and improve very simple JavaScript programs;
- can interpret and handle basic exceptions related to errors in program execution;
- understands a programmer's work in the software development process, and the role of fundamental development tools.

MANEESH GOWTHAM CHALLA

Student

Maciek Wichary
VP & CEO, OpenEDG

11 Apr 2024

Date

www.netacad.com | www.javascriptinstitute.com

Job/Internship Applied Successfully...!

Inbox

no-Reply 14/12/2023

Dear KALI VARA PRASAD, Thank you for expressing your interest in the 6months

no-Reply 14/12/2023

to me

Dear KALI VARA PRASAD,

Thank you for expressing your interest in the 6months Global Virtual internship opportunity for Data Analytics Position at 360 DIGI TMG and would like to inform you that your application has been assigned the following Application ID: 4175407.

Show quoted text

Reply Reply all Forward

No add-ons available for this email

Dileep Bhaskaran <dileep.bhaskaran@360digitmg.com>

Tue 2/27/2024 5:20 PM

Dear Participant,

We would like to inform you that this initiative will be conducted as a CSR (Corporate Social Responsibility) activity by 360DigiTMG and APSCHE, You are welcome to join the WhatsApp community for further updates.

Batch Start Date:01/03/2024
Whatsapp link : https://chat.whatsapp.com/CvBOTenPgj1BjxwX4C3R0V

Thank you for your interest and participation.

--

Regards,
Dileep Bhaskaran
Co-ordinator
Mobile:- +91-7904755532
Email:- dileep.bhaskaran@360digitmg.com

We Provide: Data Science | Artificial Intelligence | MLOps | ML on Cloud | Machine Learning | Deep Learn