Nigrani - The Door Monitoring System

Sakshi Hemant Nazirkar
Department of Information Technology
VPKBIET
Baramati

Abhishek Chandu Mankar
Department of Information Technology
VPKBIET
Baramati

Devyani Rahul Pandhare
Department of Information Technology
VPKBIET
Baramati

Rohan Chandrakant Nirwan
Department of Information Technology
VPKBIET
Baramati

Under Guidance of
Mrs Kanchan Bhale
Department of Information Technology
VPKBIET
Baramati

Abstract—With increase in digitalization and automation, peo-ple look for more comfort and security. Automation is carrying out activities with minimal human activity involved in it. In automation, a door monitoring system is proposed, the person on the door will be identified and then the door will opened. This paper will use face detection and face recognition. When the camera detects face, the image frame will be captured. The detected face will be matched with the faces in the database. If a match is found the door will be opened or else the door will stay closed. This paper will be implemented using in-built python libraries.

Index Terms—Face Detection, Face recognition,Python APIs

I. INTRODUCTION

The way of living, lifestyle of people are changing. People look for comfort in their day to day activities and therefore demand for digitalization is the need of the hour,and as a result automation is the key to a technological era. One of it is video surveillance. Video Surveillance is often done where security must be there, to prevent theft, or else to monitor the work employees. Elderly people, differently-abled people or kids are most of the time at home, without anyone over to supervise them. To pro-vide them security and safety is our major concern here. This paper proposes "DOOR MONITORING SYSTEM", using face detection, face recognition and automated door locks(to open and close the door) using raspberry pi.

A data set with the faces of the permitted people is made and stored. In the face detection part, when some face is detected in front of the camera, the frame will be captured as a temporary image file. Next face recognition will be done, by comparing the temporary image file with the images stored in the data set. If a match is found, that is, the person is been successfully recognized. The signal will be given to raspberry pi,access will be granted and the door will be opened. If there is no match, access will be denied and the door will not open.

To implement this work this paper will be using python in-built libraries and APIs. Python is a huge and rich community with proper documentation and working to be used. The proposed work is less costly and affordable to working people for their household. The system is efficient and easy to use.

II. LITERATURE SURVEY

Types of machine learning algorithms:

Supervised Learning:
This algorithm consists of a target / outcome variable (or dependent variable) which is to be predicted from a given set of predictors (independent variables). Using these set of variables, generate a function that map inputs to desired outputs.

Unsupervised Learning:
In this algorithm,there is no target or outcome variable to predict / estimate. It is used for clustering population in different groups, which is widely used for segmenting customers in different groups for specific intervention. Examples of Unsupervised Learning: Apriori algorithm, K-means.

Reinforcement Learning:
Using this algorithm, the machine is trained to make specific decisions. It works this way: the machine is exposed to an environment where it trains itself continually using trial and error. This machine learns from past experience and tries to capture the best possible knowledge to make accurate busi-ness decisions. Example of Reinforcement Learning: Markov Decision Process

List of Common Machine Learning Algorithms:
Here is the list of commonly used machine learning algo-rithms. These algorithms can be applied to almost any data classification problem :

    Linear Regression
    Logistic Regression
    Decision Tree

SVM
   Naive
   Bayes kNN

Linear Regression:

It is used to estimate real values (cost of houses, number of calls, total sales etc.) based on continuous variable(s). Here, relationship is established between independent and dependent variables by fitting a best line. This best fit line is known as regression line and represented by a linear equation Y= a *X + b.

Logistic Regression:

Don't get confused by its name! It is a classification not a regression algorithm. It is used to estimate discrete values (Binary values like 0/1, yes/no, true/false) based on given set of independent variable(s) . In simple words, it predicts the probability of occurrence of an event by fitting data to a logit function. Hence, it is also known as logit regression. Since, it predicts the probability, its output values lies between 0 and 1 (as expected).

Decision Tree:

It is a type of supervised learning algorithm that is mostly used for classification problems. Surprisingly, it works for both categorical and continuous dependent variables. In this algorithm, the population is split into two or more homogeneous sets.When the dataset is in the hand, it is important to learn about it and classify it in a most optimal way. Decision tree takes decision at each point and splits the dataset . It is a top down traversal and each split should provide the maximum information. The key advantage of this technique is when the dataset is huge and the number of features is also quite high then it is important to find the best features to split the dataset in order to perform efficient and optimal classification.

SVM (Support Vector Machine):

It is a classification method. In this algorithm, each data item is plot as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. For example, if we only had two features like Height and Hair length of an individual, first plot these two variables in two dimensional space where each point has two co-ordinates (these co-ordinates are known as Support Vectors)

Naive Bayes:

It is a classification technique based on Bayes' theorem with an assumption of independence between predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelatedto the presence of any other feature. For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, a naive Bayes classifier would

consider all of these properties to independently contribute to the probability that this fruit is an apple.

kNN (k- Nearest Neighbors):

It can be used for both classification and regression problems. However, it is more widely used in classification problems in the industry. K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases by a majority vote of its k neighbors. The case being assigned to the class is most common amongst its K nearest neighbors measured by a distance function. These distance functions can be Euclidean, Manhattan, Minkowski and Hamming distance. First three functions are used for continuous function and fourth one (Hamming) for categorical variables. If K = 1, then the case is simply assigned to the class of its nearest neighbor. At times, choosing K turns out to be a challenge while performing kNN modeling.

K-NN is one of the most used classification algorithms in the machine learning techniques area.

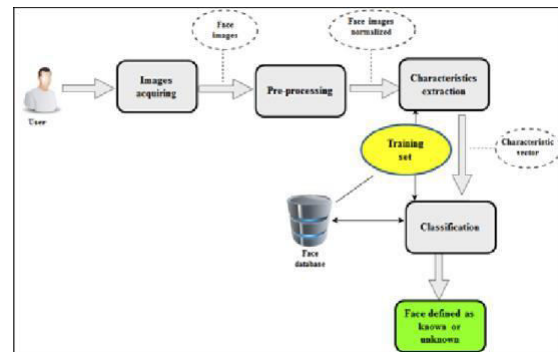## III.   PROPOSED SYSTEM

Fig. 1. Architecture

IV. MODULES

1.Valid face detection image storage:

This module mainly focuses on the detection of the face in front of the camera and their storage in the database. When some activity is detected in front o the camera, the camera searches for faces. When the valid face is detected, the camera captures the frame. The captured frame id stored in the image database. Here, image is said to be valid when proper face can be detected without any obstruction. The detected face is stored as values in a numpy array at the backend. That is, the image is not stored as an image, but encoded in numeric values. The RGB image is stored as a numpy array in the backed.

2.Face recognition and validation:

Dataset which is created in the previous module. Each RGB image is encoded in numpy array. In this module, the person to be recognized is in front of the camera. A temporary file is stored within the temporary folder. The temporary RGB image is encoded in a numpy array. The encoded temporary image

in file is matched with the database encodings. If the match is found the person is rendered valid and the access is granted to enter.
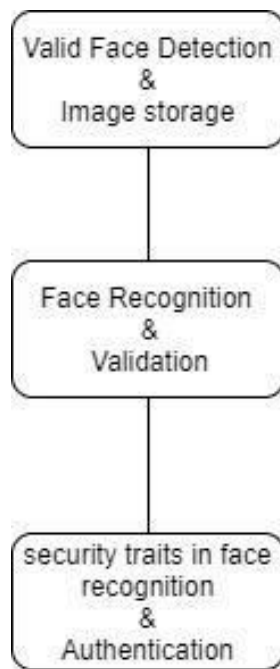

Fig. 2. Project Module

V. Libraries and Files Used

Implementation is done using Python libraries and APIs. Python is a huge and rich community, providing lots of functionality for various tasks to be carried out. Important point is to just find out the right modules, for the proper working out of the task to be performed.
The following stated modules are used for face detection and recognition in the proposed work :

Convert Text to
Speech Batch File
Numpy
Date and Time
(datetime) OS
cv2

A. Convert Text to Speech

API-win32com.Client:
There are several APIs available to convert text to speech in python. One of such APIs available in the python library commonly known as win32com library. It provides a bunch of methods to get excited about and one of them is the Dispatch method of the library. Dispatch method when passed with the argument of SAPI.SpVoice It interacts with the Microsoft Speech SDK to speak what you type in from the keyboard.

Voice Characteristics and UI Support : Use the GetVoices method to determine what other voices are available to the voice object.
Speaking and Queuing : The Speak method places a text stream in the TTS engine's input queue and returns a stream number. It can be called synchronously or asynchronously. When called synchronously, the Speak method does not return until the text has been spoken; when called asynchronously, it returns immediately, and the voice speaks as a background process.
Voice Output : An SpVoice object is created with its audio output set to the system default audio output. Use the GetAudioOutputs method to determine what other outputs are available to the voice, and use the AudioOutput property to set its audio output to one of them.

B. Batch File

Batch Scripts are stored in simple text files containing lines with commands that get executed in sequence, one after the other. Batch files use the same language as the command prompt. All you are doing is telling the command prompt what you want to input through a file, rather than typing it out in the command prompt. This saves you time and effort. It also allows you to put in some logic (like simple loops, conditional statements, etc. that procedural programming is capable of conceptually).

C. Numpy

Numpy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific com- puting with Python. Array in Numpy is a table of elements (usually numbers), all of the same type, indexed by a tuple of positive integers. In Numpy, number of dimensions of the array is called rank of the array.A tuple of integers giving the size of the array along each dimension is known as shape of the array. An array class in Numpy is called as ndarray. Elements in Numpy arrays are accessed by using square brackets and can be initialized by using nested Python Lists.

D. Date and Time

The datetime module in python works with date and time. It is a combination of a date and a time. Attributes: year, month, day, hour, minute, second.
This module supplies classes for manipulating dates and times in both simple and complex ways.

Why datetime?
The datetime module will help us to save the captured image of the video frame of that instance with the time and date

values. Thus this captured image file name can be used for further references.

E. OS

The OS module in python provides functions for interacting with the operating system. OS, comes under Pythons standard utility modules. This module provides a portable way of using operating system dependent functionality. The OS and os.path modules include many functions to interact with the file system.

F. cv2

OpenCV (Open Source Computer Vision Library) is an open source computer vision OpenCV-Python is the Python API of OpenCV. It combines the best qualities of OpenCV C++ API and Python language. The cv2.imread() function is used to read an image. cam.read() returns a bool (True/False). If frame is read correctly, it will be True. So you can check end of the video by checking this return value. cv2.VideoCapture(), function used to capture the image of that instance frame.

VI. CONCLUSION

Automation is the need of an hour, and people look for comfort in their daily activities. This paper concludes that automation combined with IoT reduces expenditure, as this paper uses python libraries, which are open source. In this paper, we implement face detection and face recognition using python libraries. Thus, overall cost and efficiency of the system is good, and also affordable to common man.

VII. REFERENCES

REFERENCES

[1] Smart home automation with a unique door monitoring system for old age people using Python, OpenCV, Android and Raspberry pi, Inter-national Conference on Intelligent Computing and Control Systems, ICICCS 2017

[2] Facial Recognition System Based on Techniques of Principal Component Analysis and Autofaces with K-NN, K-Star and Random Forest Classi ers Fbio Abrantes Diniz,Francisco Milton Mendes Neto,Francisco das Chagas Lima Jnior, Laysa Mabel de Oliveira Fontes, 2017.

[3] Jan Erick Solem. Programming Computer Vision with Python, SPD O'REILLY, 2018

[4] Preateek Joshi. Python Machine Learning Cookbook, PACKT PUB-LISHING, 2016

[5] https://towardsdatascience.com/types-of-machine-learning-algorithms-you-should-know-953a08248861 (Answered by : David Fumo)

[6] https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/

[7] https://pypi.org/project/face-recognition/