Login Portal With Facial Recognition

Kavya DN[1], Advith S Karanth[2], Rakshith Bhat[3], Amith G[4]
[1, 2, 3, 4] Dept of CSE
[1, 2, 3, 4] Atria Institute of Technology

Abstract- *Data security is a very crucial aspect in today's world. Through this project, we have designed a Dual-Factor Authentication login system. This process is done to better protect both the user's credentials and the resources the user can access. Two-factor authentication provides a higher level of security than authentication methods that depend on single-factor authentication. In this system, the first level is a password based mechanism and the second is a very basic form of face recognition that has been implemented using the Haar Cascades Classifier, OpenCV & K-Nearest Neighbors Algorithm.*

Keywords- Dual-Factor Authentication, Haar Cascades Classifier, K-Nearest Neighbors Algorithm, OpenCV.

I. INTRODUCTION

The face is the most crucial entity for human identity. It is the feature that best distinguishes a person. And for the very same reasons, Face Recognition is an important technique. Face recognition is an interesting and challenging problem and impacts important applications in many areas such as identification for law enforcement, authentication for banking and security system access, and personal identification among others.

Face recognition is an easy task for humans but it is an entirely different task for a computer. A very little is known about human recognition to date on how do we analyze an image and how does the brain encode it and are inner features (eyes, nose, mouth) or outer features (head shape, hairline) used for successful face recognition? Neurophysiologist David Hubel and Torsten Wiesel have shown that our brain has specialized nerve cells responding to specific local features of a scene, such as lines, edges, angles or movement. Since we don't see the world as scattered pieces, our visual cortex must somehow combine the different sources of information into useful patterns. Automatic face recognition is all about extracting those meaningful features from an image, putting them into a useful representation and performing some classifications on them.

The whole process can be divided into three major steps where the first step is to find a good database of faces with multiple images for each individual. The next step is to detect faces in the database images and use them to train the face recognizer and the last step is to test the face recognizer to recognize faces it was trained for.

In pattern recognition, the *k-nearest neighbors algorithm* (k-NN) is a non-parametric method. It is used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression:

In k-NN classification, the output is a class membership. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor.

In k-NN regression, the output is the property value for the object. This value is the average of the values of k nearest neighbors.

The nearest neighbor classifier works based on a simple nonparametric decision. Each query image I_q is examined based on the distance of its features from the features of other images in the training database. The nearest neighbor is the image which has the minimum distance from the query image in the feature space. The distance between two features can be measured based on one of the distance functions such as, city block distance d_1, Euclidean distance d_2 or cosine distance d_{cos}:

$$d_1(x, y) = \sum_{i=1}^{N} | x_i - y_i | \quad (1)$$

$$d_2(x, y) = \sqrt{\sum_{i=1}^{N} | x_i - y_i |} \quad (2)$$

$$d_{cos}(x, y) = 1 - \frac{\vec{x} . \vec{y}}{| x | . | y |} \quad (3)$$

K nearest neighbor algorithm uses K closest samples to the query image. Each of these samples belongs to a known class Ci. The query image Iq is categorized to the class CM which has the majority of occurrences among the K samples. The performance of the kNN classifiers highly related to value of the k, the number of the samples and their topological distribution over the feature space.

When the input data to an algorithm is too large to be processed and it is suspected to be redundant (e.g. the same measurement in both feet and meters) then the input data will be transformed into a reduced representation set of features (also named features vector). Transforming the input data into the set of features is called feature extraction. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input. Feature extraction is performed on raw data prior to applying k-NN algorithm on the transformed data in feature space.

An example of a typical computer vision computation pipeline for face recognition using k-NN including feature extraction and dimension reduction pre-processing steps (usually implemented with *OpenCV*):

1) *Haar face detection*
2) Mean-shift tracking analysis
3) PCA or Fisher LDA projection into feature space, followed by k-NN classification

II. LITERATURE SURVEY

Automated face recognition was developed in 1960s. The first semi-automated system for face recognition required the administrator to locate features such as eyes, ears, nose, and mouth on the photographs before it calculating the distances and the ratios to a common reference point, which were then compared to the reference data.

In 1970s, the problem with both of these early solutions was that the measurements and locations were manually computed. In 1990, Kirby and Sirovich applied Principal Component Analysis, a standard linear algebraic technique, to the face recognition problem. This was considered as a milestone. It is shown that less than one hundred values were required to accurately code a suitably aligned and normalized face image.

As a result of many studies, scientists come up with the decision that face recognition is not like other object recognition. Face recognition is one of the few biometric methods that possess the merits of both high accuracy and low intrusiveness. It has the accuracy of a physiological approach without being intrusive. For this reason, since the early seventies, face recognition has drawn the attention of researchers in fields from security, psychology and image processing.

Early face recognition algorithms used simple geometric models, but the recognition process has now matured into a science of sophisticated mathematical representations and matching processes. Since the early 1950s when digital computers were born and the world gained significant processing power, computer scientists have endeavored in bringing thought and the senses to the computer. During the 1980s, work on face recognition remained largely dormant. Plagued with the fears expressed in George Orwell’s 1984, most members of society are very concerned about the use of a computer system which is capable of recognizing them wherever they go. Since the 1990s, the research interest in face recognition has grown significantly as a result of the following facts:

1. The increase in emphasis on civilian/commercial research projects, the re-emergence of neural network classifiers with emphasis on real time computation and adaptation.
2. The availability of real time hardware.
3. The increasing need for surveillance related application.

In the year 1991, Turk and Pentland discovered that while using the Eigen faces techniques, the residual error could be used to detect faces in images, a discovery that enabled reliable real-time automated face recognition systems. This demonstration initiated much-needed analysis on how to use the technology to support national needs while being considerate of the public’s social and privacy concerns.

Critics of the technology complain that the London Borough of Newham scheme has, as of 2004, never recognized a single criminal, despite several criminals in the system's database living in the Borough and the system having been running for several years. "Not once, as far as the police know, has Newham's automatic facial recognition system spotted a live target." This information seems to conflict with claims that the system was credited with a 34% reduction in crime, which better explains why the system was then rolled out to Birmingham also.

In 2006, the performance of the latest face recognition algorithms was evaluated in the Face Recognition

Grand Challenge (FRGC). High-resolution face images, 3-D face scans and iris images were used in the tests.

The results indicated that the new algorithms are 10 times more accurate than the face recognition algorithms of 2002 and 100 times more accurate than those of 1995. Some of the algorithms were able to outperform human participants in recognizing faces and could identify even identical twins.

Tolba et al (2006) have reported an up-to-date review of major human face recognition research in "Face recognition: a literature review, methods and technologies of face recognition". A literature review of the most recent face recognition techniques is presented. Description and limitations of face databases which are used to test the performance of these face recognition algorithms are given.

This face recognition problem is made difficult by the great variability in head rotation and tilt, lighting intensity, angle, facial expression, aging etc. Some other attempts at facial recognition by machine have allowed for little or no variability in these quantities. Yet, the method of correlation or pattern matching of unprocessed data, which is often used by some researchers, is certain to fail in cases where the variability is great. In particular, the correlation is very low between two pictures of the same person with two different head rotations.

Modern face recognition has reached an identification rate greater than 90% with well-controlled pose and illumination conditions. The task of recognizing faces has attracted much attention both from Neuro-scientists and from computer vision scientists. While network security and access control are it most widely discussed applications, face recognition has also proven useful in other multimedia information processing areas.

III. PROPOSED SYSTEM

The system we designed comprises of and works in the manner mentioned below.

A. Technology Stack

- ✓ Python – The whole code has been written in python.
- ✓ Cv2 – cv2 is the OpenCV module and is used here for reading and writing images, also to input a video stream.
- ✓ Algorithm – KNN
- ✓ Classifier – Haar Cascades.

B. Implementation

I. **Generating Training Data**

The following steps are followed to generate training data:

- ➢ Write a Python Script that captures images from your webcam video stream.
- ➢ Extracts all Faces from the image frame (using haar cascades).
- ➢ Stores the Face information into numpy arrays.
- ➢

1. Read and show video stream, capture images
2. Detect Faces and show bounding box (haar cascade)
3. Flatten the largest face image(gray scale) and save in a numpy array
4. Repeat the above for multiple people to generate training data

II. **Building The Face Classifier**

Recognise Faces using the classification algorithm — KNN.

1) Load the training data (numpy arrays of all the persons)
   - o x- values are stored in the numpy arrays
   - o y-values we need to assign for each person
2) Read a video stream using OpenCV.
3) Extract faces out of it.
4) Use KNN to find the prediction of face (int).
5) Map the predicted ID to name of the user.
6) Display the predictions on the screen — bounding box and name.

IV. METHODOLOGY

A. Overview

k- Nearest Neighbor:k-NN is one of the foremost basic classification algorithms in machine learning. It belongs to the supervised learning class of machine learning. k-NN is usually employed in search applications wherever you're looking for "similar" things. The way we measure similarity is by making a vector illustration of the things, and then compare the vectors using an acceptable distance metric (like the geometrician distance, for example).

It is typically utilized in data processing, pattern recognition, recommender systems and intrusion detection.

Dataset used: We used haarcascade_frontalface_default.xml dataset.

Face-Recognition : This includes 3 Python files where the primary one is employed to detect the face and storing it in a list format, second is employed to store the info in '.csv' file format and the third one is employed recognize the face.

Face recognition is often described as a process that first involves four steps; they are: face detection, face alignment, feature extraction, and finally face recognition.

a. Face Detection. Locate one or more faces in the image and mark with a bounding box.
b. Face Alignment. Normalize the face to be consistent with the database, such as geometry and photometrics.
c. Feature Extraction. Extract features from the face that can be used for the recognition task.
d. Face Recognition. Perform matching of the face against one or more known faces in a prepared database.

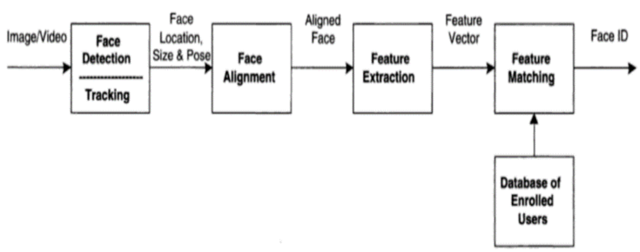

Fig 1. Face Recognition processing flow

Face Detection Task

Face detection is the non-trivial first step in face recognition.

It is a problem of object recognition that requires that both the location of each face in a photograph is identified (e.g. the position) and the extent of the face is localized (e.g. with a bounding box). Object recognition itself is a challenging problem, although in this case, it is similar as there is only one type of object, e.g. faces, to be localized, although faces can vary wildly.Further, because it is the first step in a broader face recognition system, face detection must be robust. For example, a face cannot be recognized if it cannot first be detected. That means faces must be detected with all manner of orientations, angles, light levels, hairstyles, hats, glasses, facial hair, makeup, ages, and so on.

a taxonomy of face detection methods that can be broadly divided into two main groups:

- Feature-Based.
- Image-Based.

The feature-based face detection uses hand-crafted filters that search for and locate faces in photographs based on a deep knowledge of the domain. They can be very fast and very effective when the filters match, although they can fail dramatically when they don't, e.g. making them somewhat fragile.

Alternately, image-based face detection is holistic and learns how to automatically locate and extract faces from the entire image. Neural networks fit into this class of methods.

Face Recognition Tasks

The task of face recognition is broad and can be tailored to the specific needs of a prediction problem.

For example, in the 1995 paper titled "Human and machine recognition of faces: A survey," the authors describe three face recognition tasks:

- **Face Matching**: Find the best match for a given face.
- **Face Similarity**: Find faces that are most similar to a given face.
- **Face Transformation**: Generate new faces that are similar to a given face.

The 2011 book on face recognition titled "Handbook of Face Recognition" describes two main modes for face recognition:

- **Face Verification**. A one-to-one mapping of a given face against a known identity (e.g. *is this the person?*).
- **Face Identification**. A one-to-many mapping for a given face against a database of known faces (e.g. *who is this person?*).

We can describe the problem of face recognition as a supervised predictive modelling task trained on samples with inputs and outputs.In all tasks, the input is a photo that contains at least one face, most likely a detected face that may also have been aligned.

The output varies based on the type of prediction required for the task; for example:

- It may then be a binary class label or binary class probability in the case of a face verification task.
- It may be a categorical class label or set of probabilities for a face identification task.

- It may be a similarity metric in the case of a similarity type task.

B. Approach

Step 1: Detect Face

First of all, we'll generates face patterns based on the HOG algorithmic program.

We will notice the a part of the simplified pictures that look the most similar to an original known HOG face pattern.

Finally, a bounding box is drawn around the detected face

Step 2: Get 68 points and an adjusted face

The face landmark estimation algorithmic program are going to be accustomed to figure out sixty eight specific points that exists on each faces.

From the found landmarks, OpenCV 's affine transformation can use some basic image transformations like rotation, scale and shear to try to create the eyes and lip always appear in the same location on every image.

Step 3: Get 128 measurements

The targeted face pictures are passed through a deep convolution neural network to get 128 measurements that is 128 dimensional unit hyper-sphere.

Step 4 : Apply Machine Learning algorithmic program

In this final step, apply your favourite algorithmic program for clustering, similarity detection, classification. Since we are using face recognition, classification is our path.

K-Nearest Neighbor

- KNN algorithmic program is among one of the only algorithmic program for regression and classification in supervised learning.
- KNN is non-parametric which suggests it doesn't create any assumptions however bases on the model structure generated from the data.
- KNN is termed memory-based or lazy learning as a result of the way it learns is just storing the representations of the training examples.
- An object is classified based on the majority votes of its neighbors (the training set). The new example object are going to be assigned to the category with its most similar k nearest neighbors.

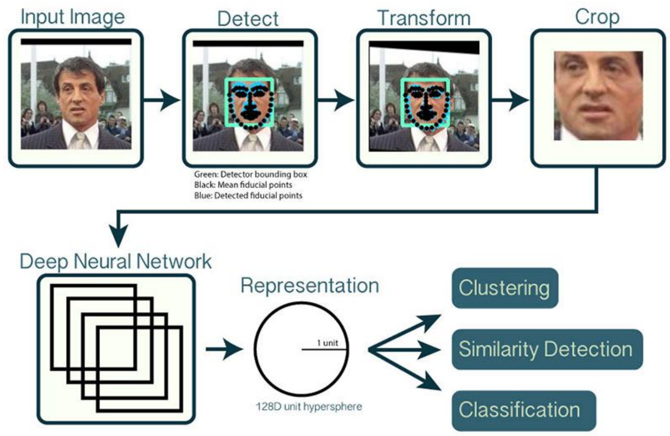

Fig 2. How OpenCV's face recognition works

V. FEASIBILITY STUDY& RESULTS

To examine the performance of the kNN classifiers and its relation to value of the k, the number of the samples and their topological distribution over the feature space a series of tests has been conducted. Table 1 shows the results of 90 experiments over a 2D feature space. For each experiment a random distribution of the samples over the N= {2, 20} number of clusters is generated. The number of clusters N is an argument of the complexity of the distributions of samples. The number of samples in this experiment varies between 10, 100 and 1000 samples. The value of k is selected from the set S={1,3,5,15,51}. Fig. 3 illustrates the distribution of 10,100 and 1000 random samples over a 2D feature space and the result of a kNN classifier. The Error rate in this example is 1.87%. As the Table.1 shows the performance of the system is rapidly increased if a large number of the samples are used. However in the most cases, increase in the number of the data is impossible.

Table I. Error rates for SVM, kNN and maximum entropy models (IIS).

		Error Rate %
SVM		7.5
maximum entropy models (IIS)		7.5
kNN	K=1	2.5
	K=3	5
	K=5	12.5

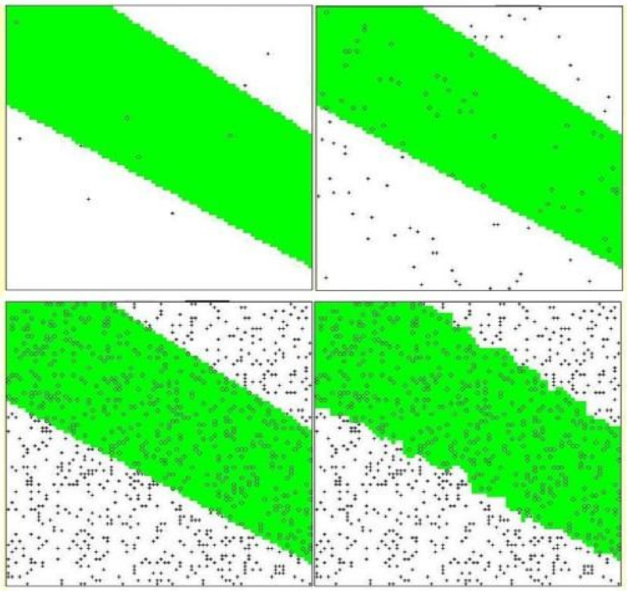

Fig 3. A(top left), B(top right) and C(bottom left) illustrate the distribution of 10,100 and 1000 random samples over a 2D feature space. The result of a kNN classifier is shown on D (bottom right) Error = 1.87%.

In order to demonstrate the performance and capabilities of this system, a face database is used to test the system. In this database, there are a number of subjects and each subject consists of certain images with variations in scale, orientationas it illustrated in the figure. The first image of each subject is used as the test image while the other 9 images are used as training set. An example face database is displayed below :

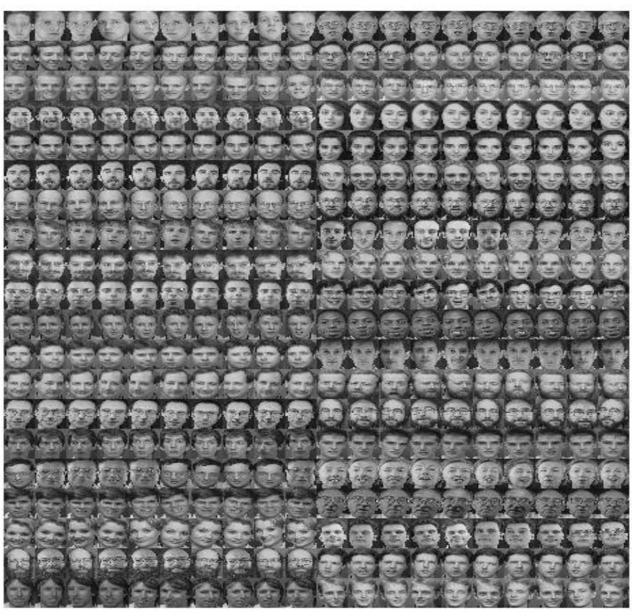

Fig 4. A face database

VI. CONCLUSION

In this paper we have presented an experiment for face identification using the KNN method. KNN is one of the simplest algorithms that can be used for classification. The face identification using the KNN method consists of two stages, such as the training phase and the testing phase. Based on the results by changing the parameter k value obtained results are different for each parameter. The results give accuracy 81% for k=1, give accuracy 53% for k=2, give accuracy 47% for k=3. From the results, it shows that the value of k greatly affects the level of accuracy of the system. The parameter k value and accuracy are inversely proportional, the greater k value gives the smaller accuracy of the identification system. From this research can be concluded that the higher k value is the smaller accuracy that we get for face identification using KNN.

VII. ACKNOWLEDGEMENT

We take this opportunity to thank our Head of the Department Dr. Aishwarya P, guideMs. Kavya DN, for their valuable guidance and for providing all the necessary facilities, which were indispensable in the completion of this paper. We are also thankful to all the Staff members of the Department of Computer Science and Engineering of Atria Institute of Technology for their valuable time, support, comments, suggestions and persuasion.

REFERENCES

[1] Biometrics Institute. (2017). Kairos- Protect your business and create frictionless customer experiences with face recognition. Retrieved from https://www.kairos.com/face-recognition-for-security

[2] Ding, C., Xu, C., & Tao, D. (2015). Multi-Task Pose-Invariant Face Recognition. IEEE Transactions on Image Processing, 24(3), 980 - 993.

[3] Zhang, X., Gonnot, T., &Saniie, J. (2017). Real-Time Face Detection and Recognition in Complex Background. Journal of Signal and Information Processing, 8(2), 99-112. Scientific Journal of Informatics, Vol. 6, No. 2, Nov 2019 159

[4] Ranjan , R., Bansal, A., Zheng, J., Xu, H., Gleason, J., Lu, B., et al. (2015). A Fast and Accurate System for Face Detection,. JOURNAL OF LATEX CLASS, 14(8).

[5] Mantoro, T., Ayu, M. A., &Suhendi. (2018). Multi-Faces Recognition Process Using Haar Cascades and Eigenface Methods. 2018 6th International Conference on Multimedia Computing and Systems (ICMCS).

[6] Rachmawanto, E. H., Anarqi, R. G., Setiadi, D. I., & Sari, A. (2018). Handwriting Recognition Using Eccentricity and Metric Feature Extraction Based on K-Nearest Neighbors. 2018 International Seminar on Application for

Technology of Information and Communication (iSemantic).

[7] Fransiska, R. (2016). Voice Recognition Using k Nearest Neighbor and Double Distance Method. 2016 International Conference on Industrial Engineering, Management Science and Application (ICIMSA).

[8] Asana, I., Widyantara, I., Wirastuti, N., &Adnyana, I. (2017). Metode Contrast Stretching untukPerbaikanKualitas Citra pada Proses Segmentasi Video. OJS Unud, 16(2).

[9] Natesan, M., &Bhuvaneshwaran, K. (2014). Image Enhancement Using Multilevel Contrast Stretching and Noise Smoothening Technique for CT Images. International Journal of Scientific and Engineering Research, 5(5), 713-718.

[10] Mahmud , F., Khatun, T., Zuhori, S., Afroge, S., Aktar, M., & Pal, B. (2015). Face recognition using Principle Component Analysis and Linear Discriminant Analysis. International Conference on Electrical Engineering and Information Communication Technology (ICEEICT).

[11] Paul , L., &Sumam, A. (2012). Face Recognition Using Principal Component Analysis Method. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 1(9).

[12] Min, W., Romanova, E., Lisovec, Y., & San, A. (2019). Application of Statistical Data Processing for Solving the Problem of Face Recognition by Using Principal Components Analysis Method. IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus). Saint Petersburg and Moscow, Russia, Russia.

[13] Cover, T., & Hart, P. (1967). Nearest Neighbor Pattern Classification. IEEE Transactions on Information Theory, 13(1).