Security Systems : Facial Recognition using MATLAB

Abhishek Anand
Dept of EEE
Manipal Institute of Technology
10abhishekanand@gmail.com

***Abstract*— Biometric refers to the automatic identification of a person based on his or her physiological and individual characteristics that can be easily verified. Among the featured measures of this system are face, fingerprint, speech recognition, retinal and signature etc. To fortify the actual presence of a real trait against a fake self-generated sample biometric system is used. In this research paper, the focus is laid on basic techniques for security system. Face recognition. In face recognition, facial recognition algorithms identify facial features by extracting landmarks, or features, from an image of the subjects face. [1]We can apply it to the servo motors using train database and test database. If the data matches the train database, it sends the command to the servo motors which in turn will open the door. [2]The UI displays ACCESS ACCEPTED or ACCESS DENIED based on the recognition using test and trained databases.This system is implemented using MATLAB.**

I. INTRODUCTION

The face assumes a noteworthy part in our social intercourse in passing on character and feeling.

For the most part, there are three stages for confront acknowledgment, for the most part confront portrayal, confront location, and face ID.

Face portrayal is the principal undertaking, that is, the manner by which to demonstrate a face. The best approach to speak to a face decides the progressive calculations of recognition and recognizable proof. For the section level acknowledgment (that is, to decide if the given picture speaks to a face), a face classification ought to be described by non specific properties of all appearances; and for the subordinate-level acknowledgment (at the end of the day, which confront class the new face has a place with), definite elements of eyes, nose, and mouth must be relegated to every individual face. There are an assortment of methodologies for confront portrayal, which can be generally ordered into three classifications: format based, highlight based, and appearance-based.

The easiest layout coordinating methodologies speak to an entire face utilizing a solitary format, i.e., a 2-D exhibit of power, which is generally an edge guide of the first face picture. In include based methodologies, geometric components, for example, position and width of eyes, nose, and mouth, eyebrow's thickness and curves, confront broadness, or invariant minutes, are separated to speak to a face. Highlight based methodologies have littler memory necessity and a higher acknowledgment speed than layout based ones do. They are especially valuable for confront scale standardization and 3D head show based posture estimation. The possibility of appearance-based methodologies is to extend confront pictures onto a straight subspace of low measurements. Such a subspace is first developed by vital part investigation on an arrangement of preparing pictures, with eigenfaces as its eigenvectors. Afterward, the idea of eigenfaces were reached out to eigenfeatures, for example, eigeneyes, eigenmouth, and so on for the discovery of facial components [3]. All the more as of late, fisherface space [4] and light subspace [5] have been proposed for managing acknowledgment under fluctuating enlightenment.

Face discovery is to find a face in a given picture and to isolate it from the rest of the scene. Face distinguishing proof is performed at the subordinate-level. At this stage, another face is contrasted with confront models put away in a database and after that arranged to a known individual if a correspondence is found. The execution of face distinguishing proof is influenced by a few components: scale, posture, light, outward appearance, and camouflage. We can scale the test picture to various sizes and utilize the scaling factor that outcomes in the littlest separation to confront space. Differing postures result from the difference in perspective or head introduction. Diverse distinguishing proof calculations outline distinctive sensitivities to posture variety. To distinguish confronts in various iIluminance conditions is a testing issue for confront acknowledgment. A similar individual, with a similar outward appearance, and seen from a similar perspective, can show up significantly unique as lighting condition changes. As of late, two methodologies, the fisherface space approach [3] and the enlightenment subspace approach [5], have been proposed to deal with various lighting conditions. The fisherface technique ventures confront pictures onto a three-dimensional direct subspace in view of Fisher's Linear Discriminant with an end goal to augment between-class disperse while Iimit inside class scrambIe.

II. METHOD USED FOR FACE RECOGNITION

A. *Eigenfaces for Recognition*

In mid 1990s, M. Turk and A. Pentland have understood that a data hypothesis approach of coding and translating face pictures may give understanding into the data substance of face pictures, stressing the critical nearby and worldwide "components". Such components might be specifically identified with our instinctive idea of face elements, for example, the eyes, nose, lips, and hair. In the dialect of data hypothesis, the goal is to separate the significant data in a face picture, encode it as productively as would be prudent, and contrast one face encoding and a database of models encoded similarly. A straightforward way to deal with remove the data contained in a face picture is to by one means or another catch the variety in a gathering of face pictures, autonomous of any judgment of components, and utilize this data to encode and think about individual face pictures. In scientific terms, the goal is to discover the foremost segments of the dissemination of appearances, or the eigenvectors of the covariance grid of the arrangement of face iamges. These eigenvectors can be thought of as an arrangement of components which together describe the variety between confront pictures. Each picture area contributes pretty much to each eigenvector, with the goal that we can show the eigenvector as a kind of spooky face called an eigenface. Some of these appearances are appeared in Figure 4. Each face picture in the preparation set can be spoken to precisely regarding a direct blend of the eigenfaces. The quantity of conceivable eigenfaces is equivalent to the quantity of face pictures in the preparation set. In any case, the countenances can likewise be approximated utilizing just the "best" eigenfacesthose that have the biggest eigenvalues, and which in this way represent the most difference inside the arrangement of face pictures. The essential explanation behind utilizing less eigenfaces is computational productivity. The most significant M eigenfaces traverse a M-dimensional subspace"confront space" of every single conceivable picture. The eigenfaces are basically the premise vectors of the eigenface deterioration. Using eigenfaces was spurred by a system for effectively speaking to pictures of confronts utilizing central segment examination. It is contended that an accumulation of face pictures can be roughly reproduced by putting away a little gathering of weights for each face and a little arrangement of standard pictures. Thusly, if a huge number of face pictures can be recreated by weighted whole of a little accumulation of trademark pictures, at that point a productive approach to learn and perceive appearances may be to manufacture the trademark highlights from known face pictures and to perceive specific faces by contrasting the element weights required with (around) reproduce them with the weights related with the known people. The eigenfaces approach for face recognition involves the following initialization operations: 1. Acquire a set of training images. 2. Calculate the eigenfaces from the training set, keeping only the best M images with the highest eigenvalues. These M images define the face space. As new faces are experienced, the eigenfaces can be updated. 3. Calculate the corresponding distribution in M-dimensional weight space for each known individual (training image), by projecting their face images onto the face space. Having initialized the system, the following steps are used to recognize new face images: 1. Given an image to be recognized, calculate a set of weights of the M eigenfaces by projecting the it onto each of the eigenfaces. 2. Determine if the image is a face at all by checking to see if the image is sufficiently close to the face space. 3. If it is a face, classify the weight pattern as eigher a known person or as unknown. 4. (Optional) Update the eigenfaces and/or weight patterns. 5. (Optional) Calculate the characteristic weight pattern of the new face image, and incorporate into the known faces.

B. Calculating Eigenfaces

Let a face image (x,y) be a two-dimensional N by N array of intensity values. An image may also be considered as a vector of dimension , so that a typical image of size 256 by 256 becomes a vector of dimension 65,536, or equivalently, a point in 65,536-dimensional space. An ensemble of images, then, maps to a collection of points in this huge space. Images of faces, being similar in overall configuration, will not be randomly distributed in this huge image space and thus can be described by a relatively low dimensional subspace. The main idea of the principal component analysis is to find the vector that best account for the distribution of face images within the entire image space. These vectors define the subspace of face images, which we call face space. Each vector is of length N^2,describes an N by N image, and is a linear combination of the original face images. Because these vectors are the eigenvectors of the covariance matrix corresponding to the original face images, and because they are face-like in appearance, they are referred to as eigenfaces. Let the training set of face images be Γ_1, Γ_2, Γ_3 , , Γ_M . The average face of the set if defined by $\Psi = \frac{1}{M}\sum_{n=1}^{M}\Gamma_n$. Each face differs from the average by the vector $\Phi_n = \Gamma_n - \Psi$. An example training set is shown in Figure 1a, with the average face Ψ shown in Figure 1b. This set of very large vectors is then subject to principal component analysis, which seeks a set of M orthonormal vectors, μ_n , which best describes the distribution of the data. The kth vector, μ_k is chosen such that

$$\lambda_k = \frac{1}{M}\sum_{n=1}^{M}(\mu_k^T\Phi_n)^2 \quad (1)$$

is a maximum, subject to

$$\mu_l^T\,\mu_k = \begin{cases} 1, l = k \\ 0, otherwise \end{cases} \quad (2)$$

The vectors μ_k and scalars λ_k are the eigenvectors and eigenvalues, respectively, of the covariance matrix

$$C = \frac{1}{M}\sum_{n=1}^{M}\Phi_n\Phi_n^T = AA^T \quad (3)$$

where the matrix $A = [\Phi_1\Phi_2 \ldots \Phi_M]$. The matrix C, however, is N^2 by N^2, and determining the N^2 eigenvectors and eigenvalues is an intractable task for typical image sizes. A computationally feasible method is needed to find these eigenvectors. If the number of data points in the image space is less than the dimension of the space (M¡NxN), there will be only M-1 , rather than N^2, meaningful eigenvectors (the remaining eigenvectors will have associated eigenvalues of zero). Fortunately, we can solve for the N^2-dimensional eigenvectors in this case by first solving for the eigenvectors of and M by M matrixe.g., solving a 16 x 16 matrix rather than a 16,384 x 16,384 matrixand then taking appropriate linear combinations of the face images Φ_n . Consider the eigenvectors v_n of $A^T A$ such that $A^T A v_n = \lambda_n v_n$ (4) Premultiplying both sides by A, we have $AA^T A v_n = \lambda_n A v_n$ (5) from which we see that $A v_n$ are the eigenvectors of $C = AA^T$. Following this analysis, we construct the M by M matrix $L = A^T A$, where $L_{mn} = \Phi_m^T \Phi_n$, and find the M eigenvectors v_n of L. These vectors determine linear combinations of the M training set face images to form the eigenfaces μ_n :

$$\mu_n = \sum_{k=1}^{M} v_{nk} \Phi_k = A v_n, n = 1, \ldots\ldots, M$$

(6) With this analysis the calculations are greatly reduced, from the order of the number of pixels in the images (N^2) to the order of the number of images in the training set (M). In practice, the training set of face images will be relatively small ($M < N^2$), and the calculations become quite manageable. The associated eigenvalues allow us to rank the eigenvectors according to their usefulness in characterizeing the variation among the images.

C. Using Eigenfaces to Classify a Face Image

- Collect a set of characteristic face images of the known individuals. This set should include a number of images for each person, with some variation in expression and in the lighting (say four images of ten people, so M=40).
- Calculate the (40 x 40) matrix L, find its eigenvectors and eigenvalues, and choose the M eigenvectors with the highest associated eigenvalues (let M=10 in this example).
- Combine the normalized training set of images according to Eq. (6) to produce the (M=10) eigenfaces.
- For each known individual, calculate the class vector by averaging the eigenface pattern vectors [from Eq. (8)] calculated from the original (four) images of the individual. Choose a threshold that defines the maximum allowable distance from any face class, and a threshold that defines the maximum allowable distance from face space [according to Eq. (9)].
- For each new face image to be identified, calculate its pattern vector , the distance to each known class, and the distance to face space. If the minimum distance and the distance , classify the input face as the individual associated with class vector . If the minimum distance but , then the image may be classified as unknown, and optionally used to begin a new face class.
- If the new image is classified as a known individual, this image may be added to the original set of familiar face images, and the eigenfaces may be recalculated (steps 1-4). This gives the opportunity to modify the face space as the system encounters more instances of known faces.

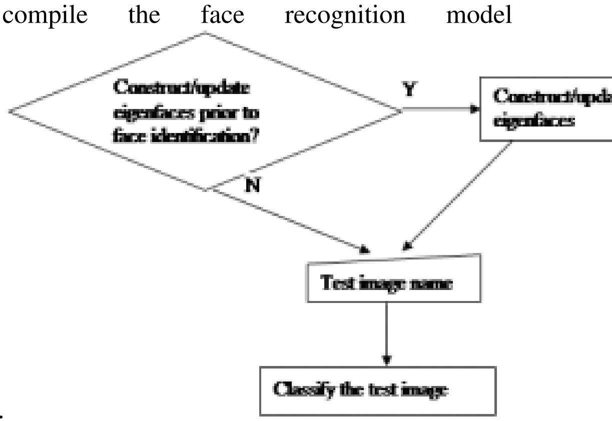
D. Implementation Issues

The whole program comprises of four fucntions, in particular 'LoadImages', 'ConstructEigenfaces', 'ClassifyNewface', and 'undoUpdateEigenfaces'. There is additionally a "main" function, which calls "ConstructEigenfaces" and "ClassifyNewface" functions to finish the face acknowledgment.

E. Functional blocks

Description of the functional blocks

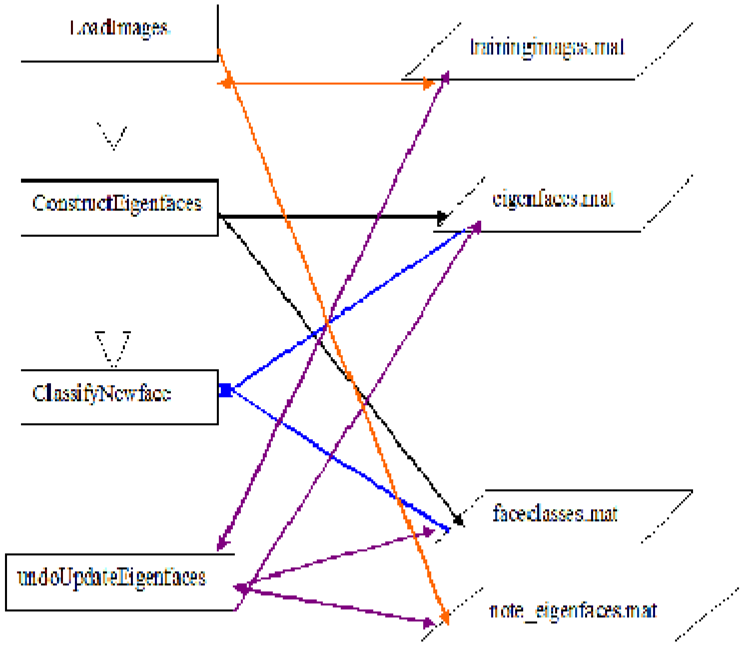
1) loadImages(ImageFileName): Loads all images meant for training and return corresponding classified values
2) contructEigenFaces(ImageFileName): Construct or Update Eigen faces and face classes.
3) classifyNewFace: Given an image input it is able to classify whether it is a face or not. Check if the face image is already in the database, if not then create new eigen face accordingly.
4) UndoUpadateEigenfaces():To undo any update actions done which shouldn't have been done.
5) main(): use all the functional blocks and compile the face recognition model



system.

F. System Structure

The structure of the system is shown in Figure 1. In the figure, the square shape indicates functions, and the parallogram represents files. An arrow pointing out from a file to a function means the function loads the file; an arrow pointing in the other direction indicates that the function creates or updates the file; a bidirectional arrow means the file is first read by the function, and later modified or updated by it. These files help the ConstructEigenfaces and ClassifyNewface functions communicate with each other in a well organized way.



III. CONCLUSIONS

An eigenfaces-based face acknowledgment approach was actualized in MatLab. This strategy speaks to a face by anticipating unique pictures onto a low-dimensional direct subspace'confront space', characterized by eigenfaces. Another face is contrasted with known face classes by processing the separation between their projections onto confront space. This approach was tried on various face pictures downloaded from various sources on the internet.

We can improve the face recognition system implemented by eigenface method. We can follow these few points for this.

1) To decrease the false-positive rate, we can influence the framework to restore various competitors from the current face classes rather than a solitary face class. Also, the rest of the work is left to human.
2) With respect to design vector speaking to a face class, we can influence each face to class comprise of a few example vectors, each developed from a face picture of a similar individual under a specific condition, as opposed to taking the normal of these vectors to speak to the face class

REFERENCES

[1] Eigenfaces for recognition, M. Turk and A. Pentland, Journal of Cognitive Neuroscience, vol.3, No.1, 1991
[2] Face recognition using eigenfaces, M. Turk and A. Pentland, Proc. IEEE Conf. on Computer Vision and Pattern Recognition
[3] Face recognition for smart environments, A. Pentland and T. Choudhury, Computer, Feb. 2000
[4] Human face segmentation and identification, S.A. Sirohey, Technical Report CAR-TR-695, Center for Automation Research, University of Maryland, College Park, MD, 1993
[5] Low dimensional procedure for the characterization of human faces, Sirovich, L. and Kirby, M, Journal of the Optical Society of America A, 4(3), 519-524