A Stochastic Technique for Face Detection System

[1]A.A. Ojugo, [1]R. Abere., [1]B.C. Orhionkpaiyo., [2]M.O. Yerokun.,
[2]A.O. Eboka and [2]I.J.B. Iyawa

[1] Department of Mathematics/Computer Sci, Federal University of Petroleum Resources Effurun
ojugo_arnold@yahoo.com, arueben2001@yahoo.com, orhionkpaiyo@yahoo.com
[4,] Computer Science Department, Federal College of Education (Technical), Asaba,
agapenexus@hotmail.co.uk, iyawaben@hotmail.com

Abstract

Biometric-based authentication systems offers several advantages over other forms of authentication methods – that has made them more popular such that the field of data security has seen a significant surge over these past few years. It is important to note that though built and designed to withstand all forms of attacks, they can still be circumvented. Thus, this study provides a means to design cum develop such system to greater withstand frequent attacks so that they can be easily deployed and employed in security-critical applications such as unattended remote applications (e-commerce, online-banking etc). The system has various strengths and limitations as noted here, as the study particularly lays emphasis on facial recognition.

Keywords: *stochastic, elitist, network, function, optimization, search space, solution, models*

1. Introduction

Personal Identification aims to associate an identity with a user – granting physical access of a system to a user. It is grouped with the various complexities into Verification (authentication) that confirm/deny a user's identity, and Recognition which establishes an identity from a set of known or unknown identity. Thus, such systems employ characteristics or physiological traits (e.g. fingerprint, facial, etc), and/or behavioral characteristics (voice, signature etc) – and are generally referred to as biometrics [6][11].

Biometrics from a soft-computing approach aims to reduce authentication task with greater certainty in time. A merit of biometrics notes that its feats for identification cannot be misplaced or forgotten – as it represents a tangible component of the user. The use of physiological or behavioral feats provides the following desirable properties [1-3]: (a) Universality – all users have the characteristic, (b) Uniqueness – no two persons have this feat, (c) Permanence – an invariant characteristic in time, (d) Collectability – quantitatively measured feat. In practice, important requirements for implementing a biometric-based authentication system are: (e) Performance – is resource requirements needed to achieve an acceptable identification accuracy, and working/environmental factors the process, (f) Acceptability – extent of user acceptability, and (g) Circumvention – how easily system is fooled.

1.1. Types of Biometric Systems

[3-4, 12-14] note biometric types as to include:

1. Voice – as not sufficiently a unique means of identification as its quality is often degraded by communication channel/device. Its signals are extracted, normalized and decomposed via frequency or time domain channels, into several band pass, as it uses techniques such as Fourier Transforms Logarithm, vector quantization, hidden Markov or dynamic time warping as the matching strategy. Input can be adversely affected by user’s health, stress and emotions; and extracting such invariant feats, in such cases can be difficult. System can be circumvented via mimicking or reproduction of a recorded voice. To combat this, system must prompt the user whose identity is to be authenticated to utter a different phrase each time.

doi: 10.4156/pica.vol3.issue1.2

2. Thermograms are images obtained via infrared radiations – with the gray levels at each pixel characterizing radiation magnitude and heat patterns that are specific to each individual. The absolute values of the heat radiations are dependent on many extraneous factors and are not completely invariant to the identity of an individual; the raw measurements of heat radiation need to be normalized with respect to heat radiating from a landmark feature of the body. The technology can be used for covert identification, distinguish identical twins and for identifying people under the influence of drugs (as the patterns contain signature of each narcotic). Its demerit are; (a) sensing issues in an uncontrolled environment with heat emanating from different spaces near the subject, drastically affects image acquisition, and (b) expensive nature of infrared sensors. It is applied in facial thermograms with non-contact and non-invasive sensing technique.
3. Fingerprints: are graphic flow-like ridges, whose formations depend on the embryonic development's initial state as they are unique to each person. As the most widely used in forensic divisions for criminal investigations, its image is captured in one-of-two ways: (i) via scanning an inked impression of a finger, and (ii) scan via a fingerprint scanner. Its representations are based either on the entire image, finger ridges, or salient feats of the ridges (minutiae). There are four (4) basic methods to fingerprint identification namely: (a) invariant feats of the gray scale profiles of fingerprint or its part, (b) global ridge patterns or fingerprint classes; (c) ridge patterns of fingerprints, and (d) fingerprint minutiae – feats from ridge endings and bifurcations.
4. Face has proven to be non-intrusive, and has two methods of image acquisition: (a) ***Transform*** – image is stored as a set of orthonormal vectors called Eigen-faces, each is a covariance analysis of the image population. Two faces are identical if they are sufficiently close in their Eigen-face feat space, and (b) ***Attribute*** – facial attributes like nose, eyes, etc. are extracted from the face image and invariance of geometric properties among in face landmark features is used for recognizing feats. A major issue is facial disguise and it is a huge task to develop recognition method that can tolerate the effects of aging, facial expressions, and slight variations in pose with respect to camera 2- or 3-D rotations).
5. Iris: Its visual texture is determined by a chaotic morphogenetic process, known to be unique for each person and each eye. Its image is captured via non-contact camera of high resolution to register the image from a predetermined distance in the camera's focal plane. The identification error is extremely small, and constant length position invariant code permits an extremely fast method of iris recognition.

Table 1. Comparison of all Biometric systems

System	Universality	Uniqueness	Performance	Acceptability	circumvent
Face	high	medium	low	high	low
Finger	medium	high	high	medium	high
Iris	high	high	High	Low	High
Retinal	high	high	High	Low	High
Voice	medium	low	Low	high	Low
Thermos	high	high	medium	high	high
Gait	medium	low	Low	High	high
DNA	high	high	High	Low	Low

Other means includes: odor, hand/finger geometry, ear, gaits, keystrokes dynamics, Deoxyribo-Nucleic Acid (DNA), signature and acoustic emissions, and retinal scan to mention a few. A comparison of all forms of biometric system is as below in table 1.

1.2. Pattern Recognition

Machine-automated recognition, description, classification and grouping of patterns is an important task in different fields such as biology, medicine, computer vision, artificial intelligence and remote sensing. A pattern is an image, handwritten word or speech signal. Its classification is into one-of-two forms: (a) **Supervised** – via discriminant analysis, where input patterns are identified, as members of a predefined class, and (b) **Unsupervised** – in clustering with patterns assigned to unknown class [23].

Recognition is a classification task, where classes are either defined by system designer (supervised), or is learned based on similarity of patterns (unsupervised). Its application include data mining, multimedia database retrieval, financial forecasting and in biometrics. Its design is in four aspects: (a) data acquisition and preprocessing, (b) representation, (c) training and testing, and (d) decision making.

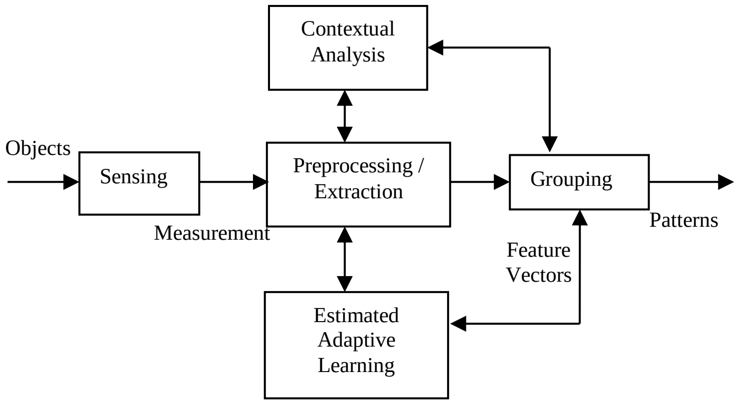

Figure 1. A Functional Pattern Recognition System

The task domain dictates sensor(s), preprocessing method, representation scheme, training and decision model. A well-defined, sufficiently constrained recognition task (with small/large inter-class variations) leads to a compact pattern representation and a simple decision making strategy. Learning from a set of examples (training set) is an important and a desired attribute of most of such systems in contrast with systems consisting of handcrafted decision rules only.

[6-7] notes four best known approaches as:

1. Template Match – determines similarity in two entities (points, curves or shapes) of same type. A template is a 2D or prototype of the pattern to be recognized, which is matched against stored template while noting allowable pose (translation and rotation) and scale changes. Similarity correlation is optimized via the training dataset, and often, template itself is learned from it. This method is computationally demanding and its few demerits makes it not the most effective.
2. Statistical Groupings: Decision and estimation theories based on vector distributing feats – defines a family of class-conditional probability density functions ($Pr\ (x|c_i)$ – probability of feat vector x given class c_i) with feats arranged in optional order (vector) are dealt with only, not considering relations between such feats.
3. Syntactic/Structural Match is a complex method of component patterns and their relationship. Its strategy for learning defines structures. It is difficult for the model to compensate for noise.
4. Neural Networks attempts to apply the models of biological neural systems to solve practical pattern recognition problems. It helps compare models with various learning process, simulate the neural system or processes in such system, based on either direct data from statistical and geometric approach, or on a higher level symbolic data from structural approach. Template matching is then compared with learning, by storing all facts as data without understanding them.

1.3. Face Recognition Algorithms

Various algorithms have been developed, to acquire images via device of high resolution (camera) with feats like weight matrix extracted from them, and stored in template database. Obtained weight matrix is identified using same mean image as obtained in the training phase, and compared against those stored in the template to get a match (check if such image exists in database). [5] The algorithms are:

a. Principal Component Analysis determines vectors of lower dimension that best approximates to a given data by taking S-dimensional vector representation of a face in a training dataset as input, and determines a T-dimensional subspace whose basis vector is maximum corresponding to the original image (with the dimension of this new subspace usually lower than the original as t<<s). If the original image elements are considered as random variables, then principle components are along the Eigen vectors, corresponding to larger Eigen values of the correlation matrix and error minimization done in a least square .

b. Independent Component Analysis (ICA) – like PCA, extracts statistics of a random variable with its second- and higher-order dependencies input minimized, and its basis along which the data is statistically independent found.
c. Elastic Bunch Graph: Faces share similar topological structure and are represented as graphs, with nodes positioned at fiducially points (eyes, nose, etc) and edges labeled with 2D vectors. Each node is a set of 40 complex Gabor wavelength coefficients at different scales and orientations (phase, amplitude). Recognition is based on labeled graphs (nodes connected by edges; each node is a jet; while edges are distances).
d. Trace Transform: is a generalization of Randon transform for image processing (for recognizing objects under transformations via translation, rotation and scaling) to yield a trace. It computes a functional along tracing lines of an image. Different Trace transforms can be produced from an image using different trace functional.
e. Evolutionary Approach is an Eigen space-based approach that searches for the best set of projection axes in order to maximize a fitness function, measuring at the same time the classification accuracy and generalization ability of the system. Because the dimension of the solution space is too big, it is involved in using a special kind of genetic algorithm called Evolutionary Pursuit.

2. Evolutionary optimization

[8] notes optimization deals with searching for optimal solution(s) in a given task, chosen from set of possible search space. It relates input and output constraints – with uncontrollable parameters and feats modeled on evolutionary, biological-inspired mathematical models that spans across various fields. Examples includes GA, GSA, ACO, PSO etc as below:

2.1. Fitness Function

A good fitness function determines if an optimal is found, as the network learns all data feats and relationships via the model that simulates future flood occurrence by computing coefficient of efficiency and other performance measures. Thus, contributes to the fitness value.

2.2. Artificial Neural Network (ANN)

[29] ANN is inspired by biological neurons or nodes (used as processing elements). A major feat is its learning ability via simulation and example, making them universal ***approximators*** as they are modeled against the human brain. Each node sends/receives electrochemical signal via ***Dendrites***, (that resends such signals to axons as converted by ***Synapse*** and thus, learning can occurs by adjusting the weights of the synapse. Synapses are connecting links, characterized by weights whose input is summed by an ***adder***. Its operation depends on task at hand, and its activation function limits its output amplitude. The synapses' effect helps modulate effects of associated inputs and the nonlinear feats exhibited by its nodes via ***transfer/activation*** function. Output is computed as weighted sum of input [14]. ***Learning*** is simply adjusting weights via algorithm that defines node's output as induced local field given by:

$$\emptyset = f(net) = f\sum_{i=1}^{m} Xi * Wij \qquad (1)$$

Encoded, ANN has three basic layers: input, hidden and output – with two configurations: ***feed-forward*** (in which signal flows from input to output without feedback and data processing extended over multiple layers of units); and ***recurrent*** (same as the feedforward but with feedback, dynamic properties and an activation values that undergoes relaxation to evolve the network to a stable state where its activation values change no more. In some tasks, its output changes significantly such that the dynamic behavior constitutes network's output). The chosen architecture is dependent on application area, feats and system requirement. ANN is configured by applying a set of inputs produces a set of desired outputs. Various methods are used to set the connection strengths namely: (a) ***explicitly*** via apriori knowledge, and (b) ***implicitly***, trains and teaches the network it patterns that changes its weight based on learning rule. Learning is divided into: ***supervised***, ***unsupervised*** and ***reinforcement*** [23].

[8-10] notes that in ***supervised*** learning, an input vector with a set of desired responses, one for each node, is relayed to output. With forward pass, errors between ***desired*** and ***actual*** response for each node in the output is found, and used to determine weight changes in learning algorithm. Thus, desired output is provided by catalyst or external teacher. E.g. backpropagation, delta and perceptron rule. In ***unsupervised*** (self organization), output is trained to respond to clusters of pattern at its input so that the system discovers statistically, salient feats of its input sample – with no prior knowledge unto which patterns are classified; Rather, the system develops its own representation of input. In ***Reinforcement***, it learns what to do, maps states to actions to help maximize a numeric reward. The network discovers what actions yield best state via trial/error. The actions may affect, not only the immediate reward, but also the next state and all subsequent rewards. These two feats, trial/error search and delayed reward are its two distinguishing properties [16-17].

2.3. Cultural Genetic Algorithm

[8-10] GA as inspired by Darwinian evolution and genetics (survival of fittest), consists of a population (data) chosen for natural selection with potential solutions to a specific task. Each potential solution is an individual/gene for which an optimal is found via four operators: initialization, selection, crossover and mutation. Individual with a gene combination close to the optimal is described as being fit. A new pool is created by mating two individuals from current pool. The fitness function is applied to determine how close an individual is to the optimal solution. GA has four steps:

a. Initialize – encodes data into format suitable for selection. Each encodings has its merit and demerit. Binary encoding is computationally more expensive to achieve. Decimal encoding allows greater diversity in chromosome and greater variance of pools generated, while float-point encoding or its combination is more efficient than binary. Thus, it encode as fixed length vectors for one or more pools of different types. The *fitness* function evaluates how close a solution is to its optimal – after which they are chosen for reproduction. If solution is found, function is *good*; else, is *bad* and not selected for crossover. The fitness function is the only part with knowledge of task. If more solutions are found, the higher its fitness value.
b. Selection – Good fit individuals close to optimal are chosen to mate. The larger the number of selected, the better the chances of yielding fitter individuals. This continues until one is chosen, from the last two/three remaining solutions, to become selected parents to new offspring. Selection ensures the fittest individuals are chosen for mating but also allows for less fit individuals from the pool and the fittest to be selected. A selection that only mates the fittest is *elitist* and often leads to converging at a local optima.
c. Crossover – ensures genes of fitter individuals are exchanged to yield a new, fitter pool. There are two crossover types (depends on encoding type used) as: (a) simple crossover for binary encoded pool via particular- or multi- point; and all genes are from one parent, and (b) arithmetic crossover allows new pool to be created by adding an individual's percentage to another.
d. Mutation alters chromosomes by changing its genes or its sequence, to ensure that a new pool converges to global minima (instead of local optima). Algorithm stops if optimal is found or after number of runs (though computationally expensive) if a number of new pools are created or once no better solution is found. Genes may change based on probability of mutation rate. Mutation improves the much needed diversity in reproduction and its algorithm is as thus:

1. Input: A chromosome rule
2. Output: Mutated solution, a fns of mutation rate
3. Set mutation threshold (between 0 and 1)
4. For each network attribute in chromosome
5. Generate a random number between 0 and 1
6. If random number > mutation threshold then

7. Generate Random value for N-Queen
8. Set solution attribute value with
9. Generated attribute value
10. End if: End For Each

Cultural GA is one of the many variants of GA with belief spaces as thus: (a) Normative (where there is a particular range of values to which an individual is bound), (b) Domain (data about task domain), (c) Temporal (data about events' space is available) and (d) Spatial (topographical data). In addition, an influence function mediates between belief space and the pool – to ensure and alter individuals in the pool to conform to belief space. CGA is chosen so as to yield a pool that does not violate its belief space and reduces number of possible individuals GA generates till an optimum is found [9].

2.4. Statement of problem and aims

Computational models do not address vital issues such as: (a) Face poses a problem as they consist of similar and same set of feats (eyes, nose and mouth etc – all with same arrangement/position, (b) many face popular detection systems do not perform well in real-time with factors such as scaling, rotation, pose and lighting. These have proven to be limiting factors in their performance, and (c) Preprocessing is required in order to obtain satisfactory results.

In most forms of crime/terrorism – it is exceedingly important to have remote sensing and monitoring systems that integrates well with security devices. Forgery and fake faces have made it expedient to have systems trained with various orientations – so that the system can arbitrarily search its database to output a user's identity based on an inputted image. The system has four modules namely: detection, alignment, feature extraction and matching.

Detection segments the face area from the background providing a coarse estimate of the localization scale of each detected face; Alignment aims to achieve more accurate localization normalizing the face from the extracted face feats, located based on location points, normalized with respect to geometrical properties, transformation and morphing such as size and pose, and improved further with respect to photometrical properties such as illumination and gray scale. After geometric and photometric normalization, extraction is done to yield effective data to help distinguish between two/more faces of different users, with respect to geometric and photometric variations. Matching helps match the extracted feats vectors of the input face to those stored in the database template to output a face's identity if match is found with sufficient confidence; Else, indicates unknown where a match is not found.

3. Methods

3.1. Network Parameter / Training

Networks are trained to perform complex functions such as recognition, classification etc. Model parameters include neurons, learning rate, and weights. In [16-17], image Eigen weights are used as inputs and its corresponding binary ID is our desired output. Training is repeated until network can identify all images in the dataset with error function reduced to acceptable value 0.2 (Figure 2). Weights and threshold obtained at training, are stored in a file to be used during recognition as in Figure 3.

With new image for recognition, its feature vectors are calculated from Eigen faces found before, and this image gets its new descriptors, as inputted to all network. Networks is then simulated with these descriptors, outputs are compared and if the maximum output exceeds the predefined threshold level, this new face is decided to belong to person with this maximum output; else, if maximum output does not exceed, face is treated as unknown [30-32].

The model adopts BP with momentum learning and hybrid with GA. BP learns weights of the multilayer network and minimize the squared error between desired output and target output, propagated back into the network as [19-22]:

a. In each output k, error $\delta_k = \delta_k \leftarrow o_k (1–o_k)(t_k - o_k)$
b. In each hidden h, error $\delta_h = \delta_k \leftarrow o_k (1–o_k) \sum w_{kh} \delta_k$, where k ϵ outputs

c. Updating the weights w_{ij}, we have that:
$\Delta w_{ij} = \eta \delta_j x_{ji}$
$W_{ji} \leftarrow W_{ji} + \Delta w_{ij}$ (where $\Delta w_{ij} = \eta \delta_j x_{ji}$)

Back-Propagation Algorithm {}

1. Initialize the weights to small random values
2. Randomly choose an input pattern x (µ)
3. Propagate the signal forward through the network
4. Compute iL in the output layer (oi = yiL)
   diL = g' (hiL) [dui-yiL], where hiL is input to the ith unit in the lth layer, and g' is activation function derivative g.
5. Compute Deltas for the preceding layer by propagating the errors backwards; δil = g'(hil) ∑wijl+1 δjl+1, for l=(L-1)..1
6. Update weights using Δwlj1 = η δil yl-1j
7. Go to step 2 and repeat for the next pattern until the error in the output layer is below pre specified threshold or a maximum number of iterations is reached.

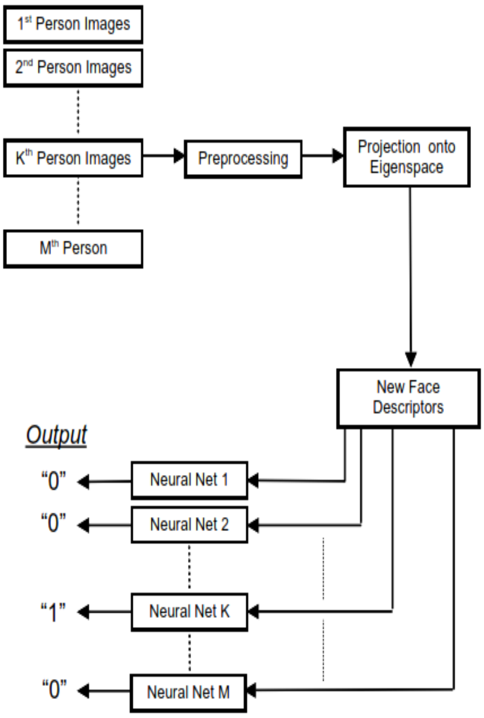

Figure 2. Training of Neural Networks

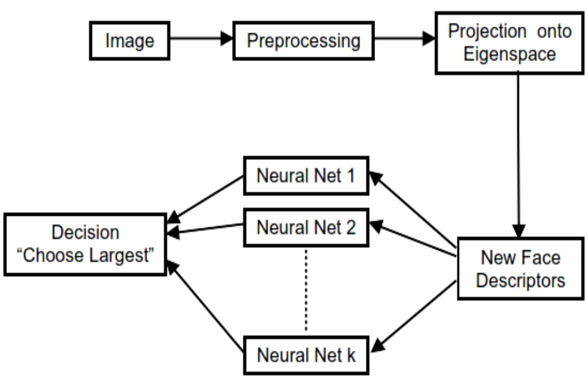

Figure 3. Simulation of Networks for Recognition

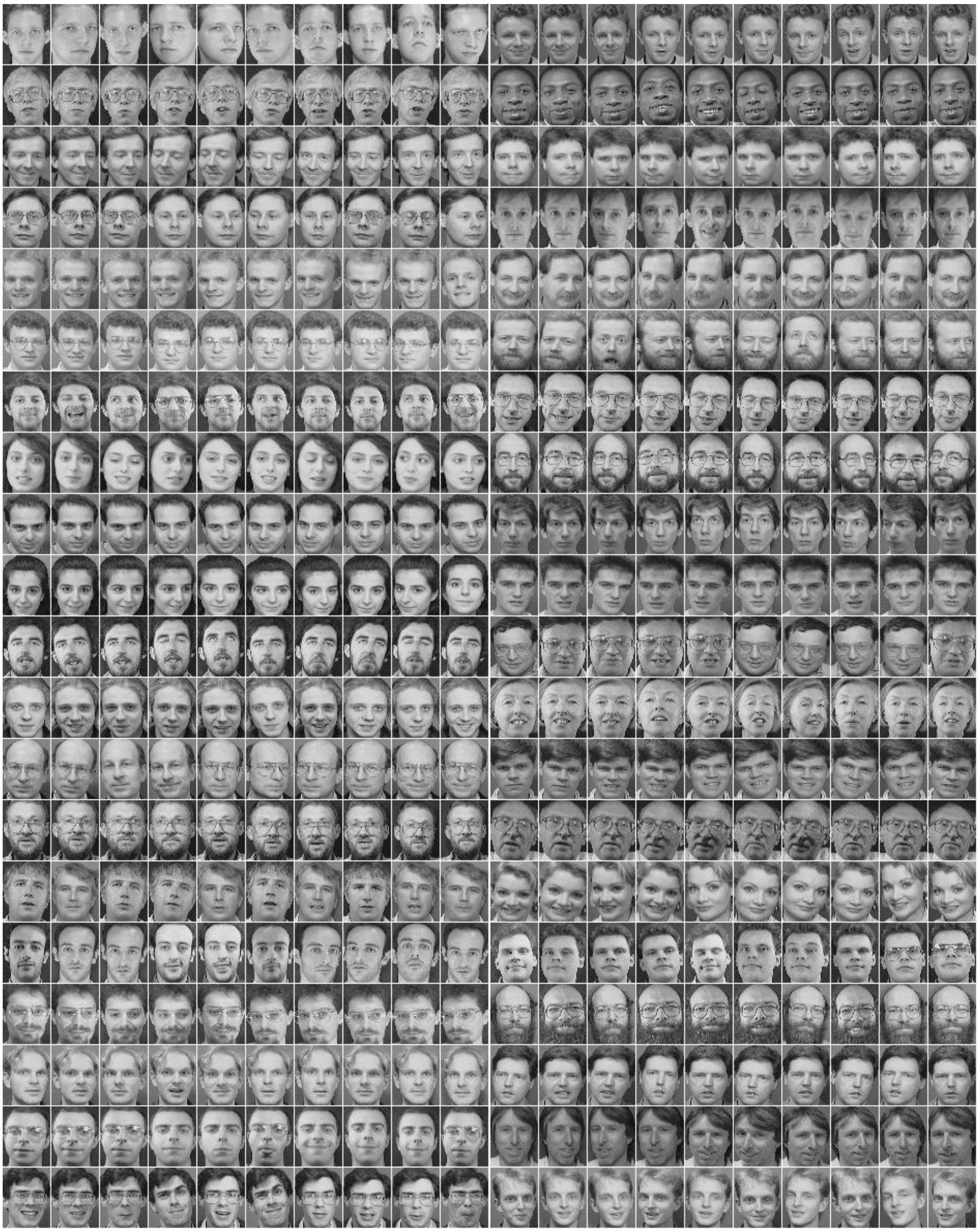

Figure 4. The Olivetti Research Laboratories image pool

Since BP is often stuck at local minima in its search where a global optima exists; But, the smaller the learning-rate parameter η is made, the smaller the changes to the synaptic weights in the network will be from one iteration to the next, and the smoother will be the trajectory in the weight space. The improvement is attained at the cost of a slower rate of learning. Else, if, learning-rate parameter is made too large in order to speed up rate of learning, the resulting large changes in the synaptic weights may yield an unstable network. Learning rate can be increased yet avoiding danger of instability by modifying the Delta rule to include a momentum as:

$$\Delta w_{ji}(n) = \alpha \Delta w_{ji}(n-1) + \eta \delta_j(n) yi(n)$$

α is positive number or momentum constant, added to cross such local optima. Once feedforward network is trained, it can be tested on a database of images. The output neuron which fires corresponds to the class of the image.

3.2. Proposed Framework / Experiment

Dataset used is the Olivetti Research Laboratory with 10 orientations of each face. The proposed framework design is grouped as:

a. Acquisition/Enhancement: Images are acquired via 480 x 640 pixels camera with proper lighting. Image is subjected to skin color algorithm, to remove background noise cum isolate skin portion from image. They are then subjected to gray scale to minimize contrasts due to lighting, texture etc, after which they are scaled so that any image taken using different capturing devices are reduced to the same resolution so that the Eigen algorithms can be efficiently applied to them. Enhancements like filtering, clipping and edge detection are made to image pool
b. Feat Extraction **–** After preprocessing, the enhanced image pass through here to find a key feat to be used for classification. Thus, module composes the feature vector that represents the image via a standard algorithm [12] as thus:
   - Assume training sets of images are $G_1, G_2 \ldots G_m$ (m = number of images).
   - Find Mean face of image: $\Psi = \left(\frac{1}{m}\right) \sum G_i$
     For i = 1 to m
   - Calculate mean–subtracted face given by:
     $F = G_i - \Psi$ for i = 1 *to* m
     Mean subtracted matrix $A = [F_1, F_2 \ldots.. F_m]$
   - Covariance matrix = $A_{mp} \times A_{pm}{}^T$
   - Find Eigen values λ_m and Eigen vectors V_{mm}
   - Eigen faces $U_k = \sum F_n V_{kn}$,$k=n=1,2,\ldots m$
   - Eigen weights $W_k = U_k T\,(G- \Psi)\ k=1,2,\ldots.m$

   The training file has ***m*** Eigen weights for each ***m*** images – inputted, with each of the image name followed by its id and Eigen weights.
c. **ANN- Learning –** Backpropagation is a supervised learning that uses derivative of the error function propagated back to contributing neurons in a network with weights updated as thus [24-28]:
   1. Set all weights to small random values.
   2. Input to each node (with x_i as input from previous node and w_i is corresponding weight, and output) is given by Sigmoid function: $Input\ \alpha = \in X_i W_i$
   $$and\ \ Output\ Y = f(x) = \frac{1}{(1+e-x)}$$
   3. The error, desired and actual output is propagated back to all nodes with weights updated via equation (w_{ij} is weight from node i to j at time t, ŋ is the learning rate and oj is output of node j and µj is error term for node j) as thus: $W_{ij}(t+1) = W_{ij} + ŋ\mu joj$
   The output node yields: $\mu j = koj(1-oj)(tj-oj)$
   Hidden nodes with μ_k as next nodal error term:
   $$\mu j = koj(1-oj) \in \mu k.wjk$$

d. Crossover/Mutation – Stored images are split into training and test dataset. Time-Lag Recurrent Network architecture (extended MLP with short-memory) with local recurrent connections is adopted as it requires a smaller network to learn temporal problems. This is because it is more plausible and computationally more powerful than other adaptive models. It uses backpropagation-in-time and momentum supervised learning for training so that its output at time t is used along with a new inputs to compute output in t+1, as in response to dynamism.

[8-10] notes the ANNGA model adopts a solution space (with 15-randomly initialized individuals, corresponding to a non-fixed face orientation) that conforms to the belief space as: (1) Normative (individuals have genes ranging from 1-to-15), (2) Domain (individual have integer as genes), (3) Spatial (individuals have integers 1-to-15 exactly once of same image with varied orientation, and (4) Temporal (mutation must not alter values of fixed faces). The *third* belief has topographic knowledge of the space (i.e. fixed values). An influence function ensures the belief space is adhered to. It also has a rounding function to ensure all values are integer and ensures the random numbers generated are not repetitions of one of the fixed numbers in the same row, column and extended diagonal [24, 26, 28].

With GA, the model is initialized with dataset whose fitness is determined and moved over from ANN training to GA via sub-pool selection of 10-images to form a new pool via *tournament* method as it is easier, more efficient to code, best suit for parallel architectures so that selection pressures are easily adjusted and it selects random numbers of individual from pool.

Algorithm: Tournament Selection {}

1. Input: Population of chromosome
2. Output: Selected Genes for crossover
3. Randomly select 3-genes from pool
4. Pick best 2-solution based on fitness value
5. Return the selected two solution
6. Do Crossover, then Select best solution as parents

Both crossover (multi-point) and mutation is applied – to distort images as randomly generated from a Gaussian distribution that corresponds to crossover points. All genes before this point, are from one parent; while the other parent contributes others. A new pool whose genes are a combination of both parents is reproduced, and they also undergoes mutation from which three random genes are selected for mutation and are allocated new random values that still conforms to belief space. New individuals replace old ones with low fitness values (creating a new pool). This continues until an individual of fitness value 0 is found – to imply solution is reached [18].

Number of mutation is applied to each pool generated, depends on how far GA has progressed (how fit is the fittest individual in the pool). Thus, knowledge of solution has direct impact on how algorithm is implemented. GA terminates when best individual has a fitness of 0 – solution is found. New individuals whose genes are a combination of both parents is produced and also undergoes mutation from which three random genes are selected for another mutation, and allocated new random values that still conforms to belief space. New individuals replace ones with low fitness values (creating new pool). This continues until a gene with fitness of 0 is found. Thus, solution is found.

Initialization/selection ensures first 3-beliefs are met; while mutation ensures the fourth belief is met. In addition, an influence function (best fitness) helps influence how many mutations takes place.

e. Testing – With GA applied to network (from model) – resulting to modified or new weights and threshold values. These are stored as used to ret-train the network via the cross-validation and testing phase. So as to help with further deformation of the images for recognition.
f. Recognition **-** The Eigen weights of the image to be identified is passed as the input to the already trained neural network and the outputs obtained. The outputs of individual neurons of the output layer are then rounded off to the nearest 0 or 1 to form a valid binary id. This binary id is then checked against the database to validate the authenticity and display the details of the face if identified.

4. Findings

With experiments, modification of the Olivetti Research Laboratories (ORL) database of faces was used with additional faces sample images. The different images for each subject provide variation in views of the individual such as lighting, facial features (such as glasses), and slight changes in head orientation etc. ORL database is used as standard set of test images used. We note that this face database contains somewhat of a bias as to the type of face representation. Males outweigh females in the subject population by a factor greater than three to one. The database was now biased with face images of Blacks, Caucasian and other races to show fair representation. Despite shortcomings in the dataset such as age representation, ORL provides a starting point for face recognition as long as we understand that test performance is an optimistic estimate [15].

Face recognition accuracy for varying number of hidden neurons is summarized in Table 2 from the same data. It is observed that for larger image sizes, there is little change in the recognition accuracy for hidden neurons more than 40. But for smaller images, this plateau is reached when the number of neurons is 20. Results show that ANN require more neurons to extract the hidden features from larger sized images. And, it is easily seen that with ANN, face recognition of more than 90% can be achieved as in Table 2 that agrees with [27-28]; while Figure 4 is the ORL image pool.

Table 2. Network Recognition Accuracy

Resolutio/Hidden Nodes	*10*	*20*	*30*	*40*	*50*	*60*
50x40	44	67	71.5	91.5	93.5	92
20x20	10	79	87.5	89	92.5	93.5
10x10	6	91.5	89.5	88	90	88.5

5. References

[1] Atick, J., Griffin, P and Redlich, A., "*Statistical approach to shape from shading: Reconstruction of 3D face surfaces from single 2D images*" Neural Computation, 1996, Vol. 8, pp. 1321-1340.

[2] Batruni, R., "*Multilayer network with piecewise-linear structure and BP-learning*", IEEE Transaction on Neural Networks, 1991, Vol. 2, pp. 395–403.

[3] Cardinaux, F., Sanderson, C and Bengio, S., "*User Authentication via Adapted Statistical Models of Face Images*", IEEE Transaction on Signal Processing, 2006, 54(1), pp. 361 - 373.

[4] Cunado, D., Nixon, M and Carter, J., "*Automatic extraction and description of gait models for recognition purposes*", Computer Vision and Image Understanding, 2003, 9(1), 1–14.

[5] Fleming, M., and Cottrell, G., "*Face categorization using unsupervised feature extraction*", International Journal of Computers and Neural Networks, 1990, 90(2), pp 99-106.

[6] Jain, A.K., Ross, A and Pankanti, S., "*Biometric: tool for data security*", IEEE Transaction Information Forensics and Security, 2006, 1(2), pp. 125–144.

[7] Marin, C., Penedo, M., Penas, M., Carreira, J and Gonzalez, F., "*Personal authentication using digital retinal images*", Springer Journal of Pattern Analysis, 2006, 9(1), pp. 21– 33.

[8] Ojugo, A.A., and Yoro, R.E., "*Computational intelligence in stochastic solution for Toroidal Queen problem",* PICA: Progress in Intelligence Computing and Applications, 2(1), doi: 10.4156/pica.vol2.issue1.4, pp 46 – 56, 2013.

[9] Ojugo, A.A., Emudianughe, J., Yoro, R.E., Okonta, E.O and Eboka, A.., "*Hybrid artificial neural network gravitational search algorithm for rainfall runoff simulation and modeling in Hydrology",* Progress in Intelligence Computing and Applications, 2(1), 2013, doi: 10.4156/pica.vol2.issue1.2, pp 22 – 33, 2013.

[10] Ojugo, A.A., Eboka, A.O., Okonta, E.O., Yoro, R.E and Aghware, F., "Genetic algorithm rule-based intrusion detection system", Journal of Emerging Trends in Computing and Information Systems, 2012, 3(8), pp 1182 - 1194.

[11] Pham, D and Karaboga, D., "Training Elman and Jordan networks for system identification using genetic algorithms", Artificial Intelligence in Engineering, 1999, 13, pp.107–117.

[12] Rizon, M and Hashim, F.,"*Face recognition using Eigen faces in neural networks*", American Journal of Applied Sciences, 2006, 3(6), pp 51 - 67.

[13] Ross, A., Dass, S and Jain, A.K., "*Deformable model for fingerprint match*", Elsevier Journal of Pattern Recog., 2005, 38(1), pp. 95–103.

[14] Sontag, E.D., "*Learning for continuous-time recurrent neural networks*", Systems and Con-

trol Letters, 1998, 34, pp. 151-158.

[15] Swets, D and Weng, J., "*Using discriminant Eigen features for image retrieval*" IEEE Transaction Pattern Analysis and Machine Intelligence, 1996, 18, pp. 831-836.

[16] Turk, M and Pentland, A., "*Eigen faces recognition*", Neuroscience, 1991, 3(1):71–86.

[17] Volkan, A., "*Face Recognition via Eigen faces in Neural Networks*", Masters' Thesis, Department Of Electrical And Electronics Engineering, The Graduate School Of Natural And Applied Sciences Of The Middle East Technical University, 2003.

[18] Weaver, A., "*Biometric Authentication*", IEEE Computer Society, 2006, 39(2), pp. 96-97.

[19] Wen, C and Ma, X., "*A max-piecewise-linear neural network for function approximation*", Neurocomputing, 2008, 71, pp. 843–852.

[20] Wright J., Yang A.Y., Ganesh A., Sastry S and Ma, Y., "*Robust Face Recognition via Sparse Representation*" Pattern Analysis and Machine Intelligence, 2009, 31(2), pp.210-227.

[21] Yun, L and Haubler, A., "*Artificial evolution of neural networks and its application to feedback control*", 1996, Artificial Intelligence in Engineering, 10(2), pp. 143–152.

[22] Zhujie, Y and Yu, Y., "*Face Recognition with Eigen face*", published Masters' thesis of the Hong-Kong University of Science and Technology, 2010, Kowloon, pp. 434-438.

[23] Kumar, S., "*Neural Networks: Classroom Approach*", McGraw-Hill, Singapore, 2005

[24] Li, S and Jain, A.K., "*Handbook of Face Recognition*", 2004, New York: Springer Verlag.

[25] Maltoni, D., Maio, D., Jain, A.K and Prabhakar, S., "*Handbook of Fingerprint Recognition*", 2003, Springer, New York.

[26] Wayman, J., Jain, A.K, Maltoni, D and Maio, D., "*Biometric Systems: Technology, Design and Performance Evaluation*", 2005, New York: Springer Verlag.

[27] Dabbah, M., Woo, W and Dlay, S., "*Secure Authentication for Face Recognition*" Proceeding of IEEE on Computational Intelligence in Image and Signal Processing, 2007, pp. 121 - 126.

[28] Dmitry, B and Starovoitov, V., "*Access Control by Face Recognition Using Neural Networks*", In Proceeding of 2nd International Conference on AI, 2002, pp 428–436

[29] Hopfield, J.J., "*Neural networks and systems with emergent collective computational abilities*", In Proceedings of National Academy of Sciences, 1979, pp.2554–2558.

[30] Phillips, P, Rauss, J and Der, S., "*Face Recognition Technology: evaluation methodology,*" Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 1997.

[31] Ross, A and Govindarajan, R., "*Feature level fusion using hand and face biometrics*", Proceeding of SPIE Conference Biometric Technology for Human Identification, 2005, pp. 196–204.

[32] Zhang, J., Yan, Y and Lades, M., "*Face Recognition: Eigen face, Elastic Matching, and Neural Nets*" Proceedings of IEEE on Memetic Algorithm, 1997 85(9):1423–1435.