ISSN 2047-3338

A Facial Recognition System Invariant to Facial Deformation Using Multi-Objective Evolutionary Granular Computing

Adegboye O. J.[1], Onifade O. F. W.[2] and Iwayemi A.[3]

[1]Department of Computer Science, The Federal Polytechnic, Ilaro, Nigeria
[2]Department of Computer Science, University of Ibadan, Nigeria
[3]Department of Computer Engineering, The Federal Polytechnic, Ile-Oluji, Nigeria
[1]olujoba.adegboye@federalpolyilaro.edu.ng, [2]fadowilly@yahoo.com, [3]iwayemi_ayodeji@yahoo.com

***Abstract*– Face recognition is a well-studied problem in which several approaches have been proposed to address the challenges of illumination, pose, expression, aging and disguise. Since these procedures modify both the shape and texture of facial features to varying degrees, it is difficult to find the correlation between pre and post-surgery facial geometry. Therefore, this work seeks to utilize image granularity yielded by multi objective evolutionary granular computing to provide resilience to variations such that local facial fragments can be used to match pre and post- surgery face images by designing and implementing an algorithm that recognizes face invariant to deformation. This research was able to use a multi-objective evolutionary granular computing based algorithm for recognizing faces altered due to deformation or reconstruction. It was done based on feature detection, face image granulation, feature descriptor extraction, and weighted chi square matching using genetic algorithm optimization. Principal Component Analysis (PCA) showed efficiency in speed while the newly developed system classified better in terms of accuracy, usage and cost. This system was able to recognize and identify faces among various dataset while achieving near good performance. It shows better performance in matching surgically altered face images against large scale gallery. With granulated information more flexibility is achieved in analyzing assimilated information from face images. Having fully explored and implemented this new system using the genetic algorithm optimization for matching the features extracted from both pre and post images, the developed system was compared with an existing system such as Principal Component Analysis (PCA) and it out-performed it in terms of cost effectiveness, accuracy and simplicity of usage.**

***Index Terms*– Algorithm, Deformation, Extraction, Face and Granulation**

I. INTRODUCTION

A) Background of the Study

The Face is a non-intrusive strong biometrics for identification. It provides information such as expression, age, gender, and identity. Hence Face recognition systems are developed to identify well informed criminals who try to hide their facial organs by different artificial means such as plastic surgery, disguise and dummy [1], [2]. Face recognition algorithm addresses two major challenges. The first is when an individual intentionally alters the appearance and features through disguise or surgery and the second is when appearance is altered through factors such as aging, variations in pose, expression, illumination and occlusion. Current face recognition systems capture faces of cooperative individuals in a controlled environment as part of the face enrollment process. It is therefore possible to control the lighting, pose, background, and quality of images; and under these conditions, the performance of face recognition is greatly enhanced. However, there is a need for more robust face recognition system for recognizing faces under unconstrained conditions such as faces that are subject to factors such as changes in pose, illumination, expression and recently introduced variations because of plastic surgery. The face recognition under unconstrained conditions results in faces, which are termed the unconstrained faces. Plastic surgery generally refers to a medical procedure that involves modifying the appearance of external anatomical features using surgical methods. Facial plastic surgeries have become increasingly popular in the recent past, especially for aesthetic improvement purposes. Some of the major facial plastic surgeries include: rhinoplasty (nose surgery), blepharoplasty (eyelid surgery), brow lift (eye-brow surgery), otoplasty (ear surgery), and rhytidectomy (face lift surgery). When an individual undergoes plastic surgery, the facial features are reconstructed either globally or locally. Therefore, in general plastic surgery can be classified into two distinct categories based on their purpose:

B) Local surgery

This is a disease correcting or reconstructive local plastic surgery. This is a kind of surgery in which an individual undergoes local plastic surgery to reconstruct the generic appearance of a facial feature, so that its functionality is restored or improved; correcting defects, anomalies, or for aesthetics such as skin texture improvement.

C) Global Surgery

Apart from local surgery, plastic surgery can be performed to completely change the facial structure which is known as full face lift. In this type of surgery the appearance, texture and facial features of an individual are reconstructed to resemble normal human face but are usually not the same as the original face. Furthermore, global plastic surgery may also be used to entirely change the face appearance, skin texture and other facial geometries making it arduous for any face recognition system to recognize faces before and after surgery. For example, restoring damaged skin due to burn injuries or accidents. The task of successfully matching face images obtained before and after plastic surgery is a challenging problem. In reality, the nature and degree to which a face is altered depends on some intersection with the variations caused due to aging, plastic surgery or disguise, and it is difficult to model such variations. Fig. 1 shows the relation among plastic surgery, aging, and disguise variations with respect to face recognition.

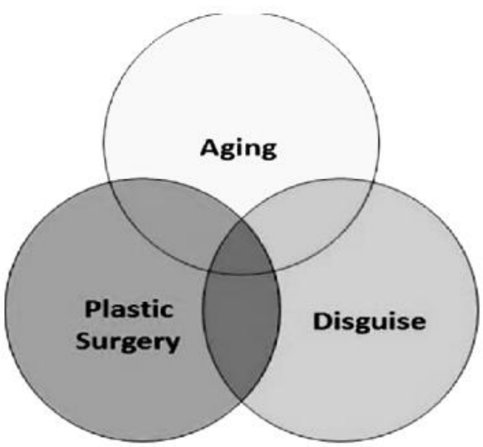

Fig. 1. Relation among plastic surgery, aging, and disguise variations with respect to face recognition [3]

II. STATEMENT OF THE PROBLEM

Face recognition is a well-studied problem in which several approaches have been proposed to address the challenges of illumination, pose expression, aging and disguise, the growing popularity of plastic surgery introduces new challenges in designing future face recognition systems. Since these procedures modify both the shape and texture of facial features to varying degrees, it is difficult to find the correlation between pre and post-surgery facial geometry. In presence of variations such as pose, expression, illumination, and disguise, it is observed that local facial regions are more resilient and can therefore be used for efficient face recognition. Therefore, this work seeks to utilize image granularity yielded by multi objective evolutionary granular computing to provide resilience to variations such that local facial fragments can be used to match pre and post- surgery face images.

The aim of this article is to design and implement an algorithm that recognizes face invariant to deformation. The objectives are:

- to improve on existing algorithms by achieving higher recognition accuracies of face under unconstrained conditions;
- to provide empirical information that will support research and help gain significant insights on the effect of deformation procedures on different facial features and their impact on existing face detection systems; and
- to evaluate the performance of the developed system with the existing ones.

III. LITERATURE REVIEW

A) Overview

This section contains the review of various aspects related to face recognition invariant to face reconstruction. An image defined in the "real world" is considered to be a function of two real variables, for example, a(x,y) with a as the amplitude (e.g., brightness) of the image at the real coordinate position (x,y). An image may be considered to contain sub-images sometimes referred to as regions–of–interest, ROIs, or simply regions. This concept reflects the fact that images frequently contain collections of objects each of which can be the basis for a region [4]. Image processing is any form of signal processing for which image is the input, such as a photograph and the image processing output may be whether an image or, a set of characteristics or parameters associated to the image [5]. There are two goals for image processing: one is to obtain the image that is more suitable for human observing and understanding, the other one is to recognize the image automatically by computer. The key step is to decompose a large and complex image into small image with independent feature [6].

From a broader definition, Digital Image Processing involves the Modification of the pixels in an image based on some function as shown in Fig. 2.

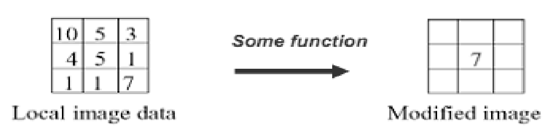

Fig. 2. Pixels Modification [7]

Usually the filters are normalized so that the sum of the entries is one. That is, the moving average filter is a convolution of the input signal with a rectangular pulse having an area of one. This avoids amplitude bias so that average gray levels are kept the same. For example, in an area of the image that has constant gray levels, the filtered image will also have this same constant gray level. If the sum were not one, then the gray level in a constant region would get multiplied by the sum of the pixels in the mask. That is, if the sum of the coefficients of the mask is one, then the average brightness of the image is not altered. If the sum of coefficients is zero, then the average brightness is lost, and it returns an image which is dark [8]. The filtered pixel value is the median of the values in a window centered on the pixel. It produces as output at a pixel the median, rather than the mean, of the pixel values in a square window centered around that pixel [8]. Fig. 3 shows the application of Laplacian to Original image.

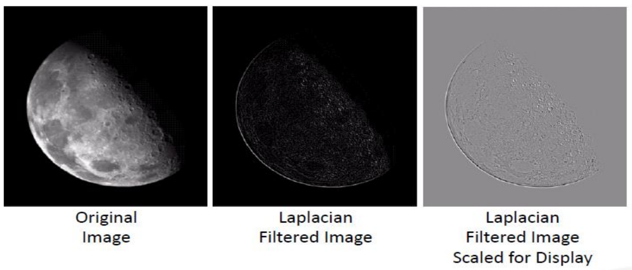

Fig. 3. Application of Laplacian to Original Image [5]

IV. RESEARCH METHODOLOGY

A) Overview

The methodology is a multi-objective evolutionary granular computing based algorithm for recognizing faces altered due to deformation or reconstruction. It is undertaken in the following sequence as shown in Fig. 4:

i). Feature detection
ii). Face image granulation
iii). Feature descriptor extraction
iv). Weighted chi square matching using genetic optimization

B) Feature Detection

A feature is a significant property of an object than can be used as part of input to processes that lead to distinguishing the object from other objects or to recognize it. It is a unique visual trait, associated to the visual primitives that constitute an object, such as edges, corners or lines among others. Thus, it is important to know which parts of the object are relevant and characteristic of it, and then describe them in a way that we can enhance the differentiable properties.

Feature Detection is a compulsory step to do in order to obtain Feature Descriptors. This family of processes locates points and regions, and they are generally capable of reproducing similar performance that the human would provide in locating elemental features in images. Since features are used as the starting point and main primitives for subsequent algorithms, the overall algorithm will be as good

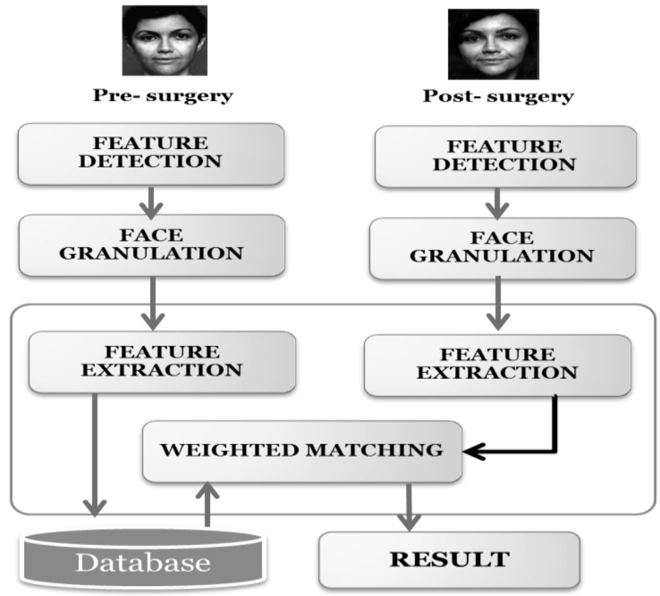

Fig. 4. Methodology

as the Feature Detector is, and hence, is the relevance of this step. Consequently, the desirable property of a Feature Detector is repeatability: whether or not the same feature will be detected in two or more different images of the same scene. Feature Detection is often a low-level image processing operation. It is usually performed as the first operation on an image, and it examines every pixel to assess the presence of a feature at that pixel.

Image Segmentation:

Image segmentation separates the objects and components of the image: It is the process of assigning label to every pixel in an image such that pixels with the same label share certain visual characteristics. It is based on discontinuity and similarity of image intensity values. There are two ways of approaching image segmentation. The first is boundary based and detects local changes. The second is region-based and searches for pixel and region similarities.

The approach here is to partition an image based on abrupt changes in intensity value such as edges. The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image. For intensity images, four popular methods are there for image segmentation which includes:

i). Threshold techniques
ii). Edge based methods
iii). Region based techniques
iv). Connectivity-preserving relaxation methods

Edge Detection:

Edge detection is by far the most effective approach for detecting meaningful discontinuities in intensity values. The points at which image brightness changes sharply are typically organized into a set of curved line segments called edges. Edge detection is a fundamental tool in image processing, machine vision and computer vision. It can help in tasks such as image retrieval, image registration, object recognition, object categorization and texture classification, among others [9]. The end usage of discovering clear and defined changes in image intensity is to represent the bottom line events and changes in the material properties of the world. Causes of Intensity alteration normally represent two types of events: one is geometric events and other is non-geometric events.

Geometric Events:

- Surface (boundary) discontinuities
- Discontinuities in depth
- Colour and Texture discontinuities

Non-Geometric Events:

- Illumination changes
- Specularities
- Shadows
- Inter-Reflections

In this work [5], edge detection is employed and it is performed in the following steps:

1. *Smoothing/Noise reduction:* To suppress as much noise as possible, without destroying the true edges.
2. *Sharpening/enhancement:* To apply a filter to enhance the quality of the edges in the image
3. *Detection:* To determine which edge pixels should be discarded as noise and which should be retained.
4. *Localization:* determine the exact location of an edge (sub-pixel resolution might be required for some applications, that is, estimate the location of an edge to better than the spacing between pixels).

The Laplacian of Gaussian filter or the Marr-Hildreth edge-detection algorithm is used and it is summarized as follows:

1. The input image is Gaussian smoothed with a n x n Gaussian low-pass filter
2. The Laplacian of the image resulting from step 1 is computed using, for example, the 3 x 3 mask:

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

3. The zero crossings of the image from step 2 is determined

The Laplacian of Gaussian ∇^2 G(x; y) kernel can be generated and used for edge detection in one convolution with the input image

$$LoG \triangleq \nabla^2 G(x,y) = \frac{\partial^2}{\partial^2 x} G(x,y) - \frac{\partial^2}{\partial^2 y} G(x,y) \quad (1)$$

To detect a zero crossing:

1. The maximum of all positive Laplacian responses and the minimum of all Laplacian responses in a 3 x 3 window is formed.
2. If all four neighbours of p, left, right, up and down have the same sign as p, then p is not a zero crossing.
3. If p has the smallest absolute value compared to the neighbours with opposite sign, the p is a zero crossing.

Fig. 5 shows the Laplacian of Gaussian Kernel.

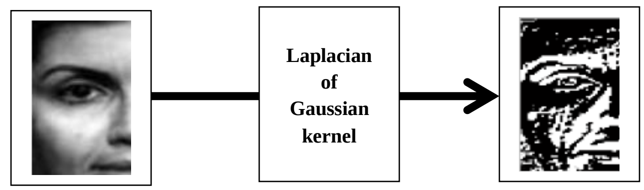

Fig. 5. Laplacian of Gaussian Kernel

C) Face Image Granulation

Face image granulation involves generating non-disjoint face granules where each granule represents different information at varying size and resolution. Let F be the detected frontal face image of size n × m. The first level provides global information at multiple resolutions as seen in Fig. 6. This is analogous to a human mind processing holistic information for face recognition at varying resolutions. Inner and outer facial information are extracted at the second level (see Fig. 7). Local facial features play an important role in face recognition by human mind. At the third level features are extracted from the local facial regions. Fig. 8 shows the Face Granules in the third level.

First level granularity	Purpose
The Gaussian and Laplacian pyramidal operators are applied to Generate a sequence of low pass filtered images in which iteratively convolving images at multiple resolutions to get a pyramidal structure.	Provide edge information Help identify plastic surgery procedure: e.g. skin lifting

Fig. 6. Face granules in the first level of granularity

Second level granularity	Purpose
Image is divided into horizontal and vertical granules.	The relation between horizontal and vertical granules is used to provide resilience to variations in inner and outer facial regions: e.g., forehead, ears, cheeks.

Fig. 7. Vertical and horizontal face granules from the second level of granularity

Third level granularity	Purpose
Generates non-disjoint face granules from local facial fragments.	To provides unique features for handling variations due to plastic surgery. To define the facial feature size

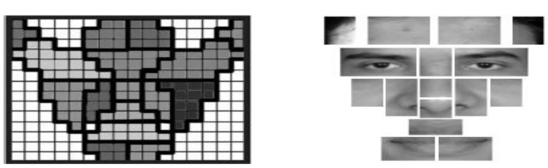

Fig. 8. Face granules in the third level of granularity

D) Feature Descriptor Extraction

The results of feature detection are a series of points, called key-points or points of interest. Once we know which points are relevant and hold the most distinctive information, it is necessary to obtain features that uniquely describe this information. This step is as necessary as the previous one and, as a result, we will get a Feature Vector which can be used in later stages in many ways: it can be compared to a series of Feature Vectors extracted from objects in a database to perform object recognition or use the information to decide, for example, if an object is a face or not depending on whether it has some compulsory features or not. The granulation processes results in granules with varying information content. Some granules contain fixed standard of reference features such as eyes, nose, and mouth while some granules predominantly contain skin regions such as forehead, cheeks, and outer facial region. Therefore, different feature extractors are needed to encode diverse information from the granules. In this work, two feature extractors are used. They are

1. Extended Local Binary Patterns (ELBP)
2. Scale Invariant Feature Transform (SIFT) are used

EXTRACTOR	FEATURE
ELBP	Binary patterns used for texture analysis.
SIFT	Sift descriptors that will be used for matching: compact representation of an image based on the magnitude, orientation, and spatial vicinity of image gradients. These descriptors are concatenated to form image signature

The LBP feature vector, in its simplest form, is created in the following manner:

1. The examined window is divided into cells (e.g., 16x16 pixels for each cell).
2. For each pixel in a cell, the pixel is compared to each of its 8 neighbors (on its left-top, left-middle, left-bottom, right-top, etc.). Following the pixels along a circle, i.e., clockwise or counter-clockwise.
3. Where the center pixel's value is greater than the neighbor's value, "1" is assigned. Otherwise, "0" is assigned. This gives an 8-digit binary number (which is usually converted to decimal for convenience).
4. The histogram is computed over the cell of the frequency of each "number" occurring (i.e., each combination of which pixels are smaller and which are greater than the center).
5. Optionally the histogram may be normalized.
6. (Normalized) histograms of all cells are concatenated. This gives the feature vector for the window.

The components of the SIFT framework for keypoint detection are as follows:

Scale-space Construction:

The first step toward the detection of interest points is the convolution of the image with Gaussian filters at different scales. It is implemented efficiently by using a difference-of-Gaussian function to identify potential interest points that are invariant to scale and orientation. Using a cascade Gaussian filtering a set of octaves is generated. At each level of the pyramid the image is rescaled (sub-sampled) and smoothed by a Gaussian. The convolved images are grouped by the octave (an octave corresponds to doubling the value of σ), and the value of k is selected so that we obtain a fixed number of blurred images per octave

1. Divide width and height by 2
2. Take average of 4 pixels for each pixel (or Gaussian blur with different σ)
3. Repeat until image is tiny
4. Run filter over each size image and hope its robust

The SIFT scale-space image representation consists of a set of N octaves $(\boldsymbol{\theta_1}, \cdots, \boldsymbol{\theta_n})$ defined by two parameters s and σ Let f be the input image. Each octave is an ordered set of s + 3 images such that

$$L(x, y, k^m\sigma) = G(x, y, k^m\sigma) \star f_i(x, y), \ k = \sqrt{2} \quad (2)$$

where,

f_i = i-th sub-sample of f

m = 0,1, …, s+2

i = 1,…, N

$$\boldsymbol{G(x, y, k^m\sigma)} = \frac{\boldsymbol{1}}{\boldsymbol{2\pi\sigma^2}} \boldsymbol{e^{-(x^2 + y^2)/2\sigma^2}} \quad (3)$$

Suppose s = 2. Then each octave contains s + 3 images.

Calculate the difference of Gaussian for Corner Detection

The keypoints extracted by SIFT are corners, i.e., discontinuity points of the gradient function. These are extracted by a DoG (difference of Gaussians). The DoG is obtained by subtraction of subsequent images in the considered octave**.**

$$DoG(x, y, \sigma, \sigma') = [G(x, y, \sigma) - G(x, y, \sigma')] \star f(x, y) \quad 4)$$

1. Take features from differences of these images-producing the gradient image
2. If the feature is repeatedly present in between Difference of Gaussians (DOG), it is scale Invariant and should be kept. The DOG/Local Extrema detection and the pixels are represented in Fig. 9 and Fig. 10 respectively;

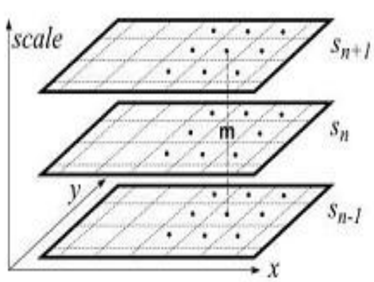

Fig. 9. DOG/Local extrema detection

The pixel marked x is compared against its 26 neighbors in a 3 x 3 x 3 neighborhood that spans adjacent DoG images [10] The keypoints are the extrema of the DoG functions, i.e., they are maximum or minimum of the function

DoG *(x, y, σ)*

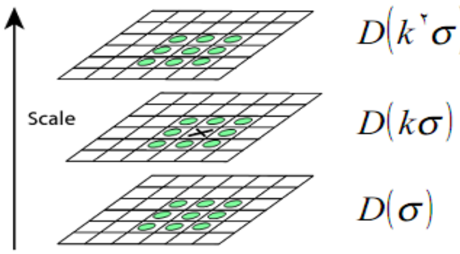

Fig. 10. Pixels

Interest points (called keypoints in the SIFT framework) are identified as local maxima or minima of the DoG images across scales. Each pixel in the DoG images is compared to its 8 neighbors at the same scale, plus the 9 corresponding neighbors at neighboring scales. If the pixel is a local maximum or minimum, it is selected as a candidate keypoint.

Sub-pixel Localization and filtering

The location of the extrema is refined by considering a parabolic fit. Due to the re-iterated Gaussian filtering, many extrema exhibit small values of the contrast. These keypoints are not robust to noise and they are generally not relevant for the description of the image. Two filters are used to discard the keypoints with small contrast and the edges that are not discriminative for the image.

This step is achieved by considering the approximation of the DoG gradient by the Taylor polynomial truncated at the first order.

For the Taylor Series Expansion

$$D(\mathbf{x}) = D + \frac{\partial D}{\partial \mathbf{x}}^T \mathbf{x} + \frac{1}{2}\mathbf{x}^T \frac{\partial^2 D}{\partial \mathbf{x}^2}\mathbf{x} \quad (5)$$

Differentiate and set to 0

$$\hat{\mathbf{x}} = -\frac{\partial^2 D}{\partial \mathbf{x}^2}^{-1} \frac{\partial D}{\partial \mathbf{x}}. \quad (6)$$

to get location in terms of (x,y,σ)

Assign Keypoints Orientations

Based on local image gradient, each keypoint is assigned a direction. In case of more strong directions, additional keypoints are created. To determine the keypoint orientation, a gradient orientation histogram is computed in the neighborhood of the keypoint (using the Gaussian image at the closest scale to the keypoint's scale). The contribution of each neighboring pixel is weighted by the gradient magnitude and a Gaussian window with a σ that is 1.5 times the scale of the keypoints. Peaks in the histogram correspond to dominant orientations. A separate keypoint is created for the direction corresponding to the histogram maximum, and any other direction within 80% of the maximum value. All the properties of the keypoint are measured relative to the keypoint orientation, this provides invariance to rotation. Create gradient histogram (36 bins) Weighted by magnitude and Gaussian window (is 1.5 times that of the scale of a keypoint). A neighborhood N around each keypoint is considered. The orientation of the gradient of the points in N is represented by a histogram H with 36 bins. The peak of H is assigned to (x, y, σ), so that the keypoint is described now by a vector (x, y, σ, θ), where θ is the orientation of the peak of H. If there are more peaks q1… qn more keypoints (x, y, s, θ), …, (x, y, s, qn) are generated.

Build Keypoint Descriptors

1. The magnitudes of all the points in the neighborhood are smoothed by a normalized Gaussian filter with σ = x.
2. The neighborhood is divided into 4×4 regions. In each region the vectors (magnitude and direction of points) are histogrammed into 8 buckets covering 360 using trilinear interpolation.
3. The feature is computed from these descriptors in the neighborhood by computing a normal of the descriptors in the neighborhood.
4. The resulting descriptor is represented as a normalized x/2 × x/2 descriptor array each with an 8 bucket histogram of vectors. Thus, the feature is $\log_2 8x^2/4 = 2 \log x+1$ bits long.

SIFT descriptors computed at the sampled regions are then concatenated to form the image signature. Similar to ELBP, weighted χ2 distance is used to compare two SIFT descriptors.

V. WEIGHTED CHI SQUARE MATCHING USING GENETIC OPTIMIZATION

The steps involved:

i). Encoding
ii). Initial population
iii). Fitness Function
iv). Mutation.

Genetic Encoding: A chromosome is a string whose length is equal to the number of face granules i.e., 40 in our case. For simultaneous optimization of two functions, two types of chromosomes are encoded: (i) for selecting feature extractor (referred to as chromosome type1) and (ii) for assigning weights to each face granule (referred to as chromosome type2). Each gene (unit) in chromosome type1 is a binary bit 0 or 1 where 0 represents the SIFT feature extractor and 1 represents the ELBP feature extractor. Genes in chromosome type2 have real valued numbers associated with corresponding weights of the 40 face granules.

Initial Population: Two generations with 100 chromosomes are populated. One generation has all type1 chromosomes while the other generation has all type2 chromosomes.

1) For selecting feature extractors (type1 chromosome), half of the initial generation (i.e., 50 chromosomes) is set with all the genes (units) as 1, which represents ELBP as the feature extractor for all 40 face granules. The remaining 50 chromosomes in the initial generation have all genes as 0 representing SIFT as the feature extractor for all 40 face granules.

2) For assigning weights to face granules (type2 chromosome), a chromosome with weights proportional to the identification accuracy of individual face granules is used as the seed chromosome. The remaining 99 chromosomes are generated by randomly changing one or more genes in the seed chromosome. Further, the weights are normalized such that the sum of all the weights in a chromosome is 1. Fig. 11 shows the Genetic optimization process for selecting feature extractor and weight for eachface granule.

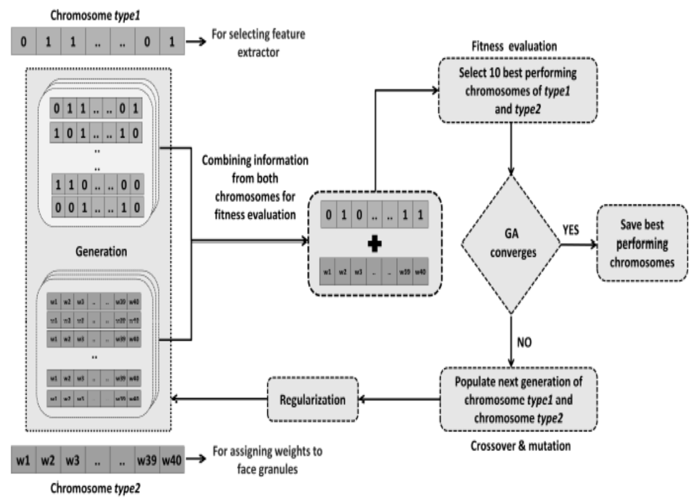

Fig. 11. Genetic optimization process for selecting feature extractor and weight for eachface granule [11]

Fitness Function: Both type1 and type2 chromosomes are combined and evaluated simultaneously. Recognition is performed using the feature extractor selected by chromosome type1 and weight encoded by chromosome type2 for each face granule. Identification accuracy, used as the fitness function, is computed on the training set and 10 best performing chromosomes are selected as parents to populate the next generation. Overall outcome is calculated based on whether a field of the connection matches the pre-classified data set, and then multiply the weight of that field. The Matched value is set to either 1 or 0. The absolute difference between the outcome of the chromosome and the actual outcome is then computed using the following equation.

$$\Delta = (actual\ outcome - expected\ threshold)$$

Once a mismatch happens, the penalty value is computed using the absolute difference

$$penalty = (\frac{\Delta * ranking}{100})$$

The ranking in the equation indicates whether or not a granule is easy to identify

Fitness = 1- penalty

The range of the fitness value is between 0 and 1

Natural Selection: The best individual (chromosome) is picked out as the final result and saved for future real life matching once the optimization meets its target. A number of individual are selected based on defined fitness function, the remaining are discarded.

Crossover: A set of uniform crossover operations is performed on parents to populate a new generation of 100 chromosomes. Crossover operation is same for both type1 and type2 chromosomes.

Mutation: After crossover, mutation is performed for type2 chromosomes by changing one or more weights by a factor of its standard deviation in the previous generation. For type1 chromosome, mutation is performed by randomly inverting the genes in the chromosome. The search process is repeated till convergence and terminated when the identification performance of the chromosomes in new generation do not improve compared to the performance of chromosomes in previous five generations. At this point, the feature extractor and optimal weights for each face granule (i.e. chromosomes giving best recognition accuracy on the training data) are obtained. Genetic optimization also enables discarding redundant and non-discriminating face granules that do not contribute much towards the recognition accuracy (i.e. the weight for that face granule is close to zero). This optimization process should lead to both dimensionality reduction and better computational efficiency.

VI. IMPLEMENTATION AND DISCUSSION

A) System Implementation

This section discusses the implementation of the system which recognizes a face even if the face is being deformed or reconstructed. The different stages discussed in the previous chapter were followed in the implementation exercise based on the hardware and software interfaces for implementation. The software application was implemented with C# programming Language with the Visual Studio 2015 version using Dot Net framework 4.0, also the image processing was aided using the OpenCV Dot Net API of the famous EmguCV software engine for image processing. The Hardware components employed are basic working Computer Hardware components with a WEBCAM peripheral. It should be noted that the main aim of the project is to design and implement an algorithm that recognizes face invariant to deformation.

B) Software Architecture (Modules)

The entire algorithm has been divided into four different modules, which allows the program to effectively carry out the expected objectives:

- Features Detection
- Feature Granulation (with the three levels of granulations)
- Feature Extraction
- Matching

C) Interface Design

This system was implemented as a real time system hence there are various window interfaces which enables the user to effectively see the method at which the system is performing the activities.

Home Page Interface: Fig. 12 shows the home page interface, it comprises three boxes namely the capture face box, the recognize face box and the storage box for captured faces.

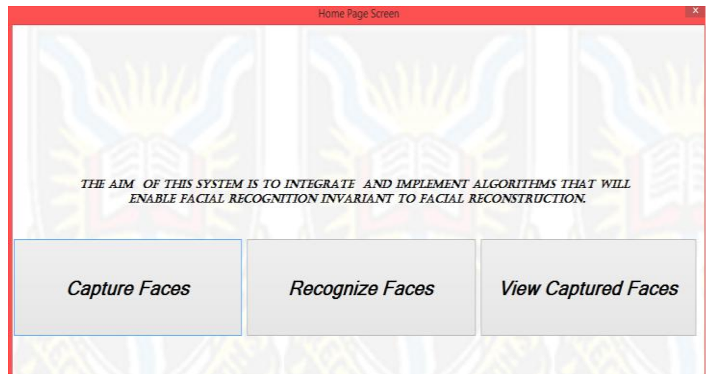

Fig. 12. Home page interface

The Capture Face Box: When this box is been clicked it will then open up another interface known as CAPTURE FACES AND SAVE WITH USERNAME / USERID.

The user's face is captured by the machine (WEBCAM), upon clicking on the start face detection box thereafter a face indicator box with a square box inscription will be displayed indicating that the face has been captured, then the user will be instructed to enter username or better still the userid, which will then allowed the face to be stored in the database. Fig. 13 and Fig. 14 show the interface, the interface bearing a face and some captured faces respectively.

Fig. 13. Capture face interface

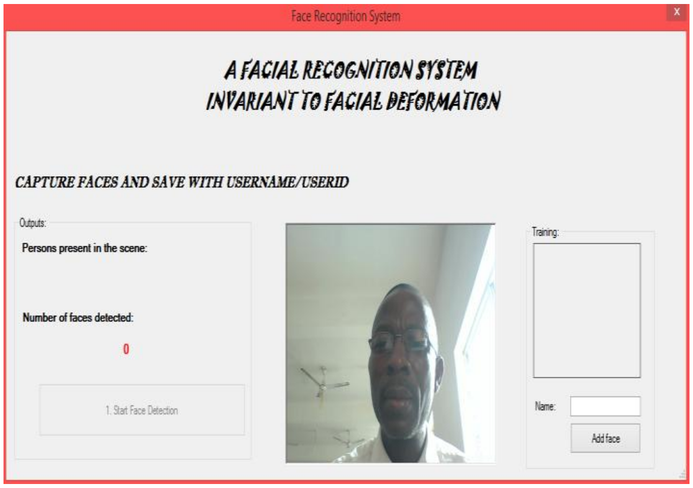

Fig. 14. An Interface showing the sample of a face of a captured user

Fig. 15. Some captured faces

D) Performance Evaluation

Performance measure and metrics were employed in order to evaluate the performance of the system; the metrics adopted include the following:

- Reliability
- Efficiency
- Timeliness
- Effectiveness
- Quality
- Usability

Some of the questions that were developed from the performance metrics include:

1. Does the system produce a good result at minimum resource cost?
2. Can the result produced be relied upon?
3. Is the time of operation alright by you?
4. Does the system meet the stated aim and objectives?
5. Are you satisfied with the operation of the system?
6. Can the system be easily used by any interested person?

The options available for the respondents (user) to choose in answering the questions are:

- Agreed
- Disagreed
- Indecisive

Questionnaires were given to peoples (respondents) whose images were captured and the results/analysis are as shown in Table 1 and represented in Fig. 16. Fig. 17 presents an interface showing a recognized face.

Table 1: Analysis of the Questionnaire

Questions	(20) Agreed	(20) Disagreed	(20) Indecisive
Does the system produce a good result at minimum resource cost?	13	5	2
Can the result produced be relied upon?	16	0	4
Is the time of operation alright by you?	6	12	2
Does the system meet the stated aim and objectives?	14	0	6
Are you satisfied with the operation of the system?	10	8	2
Can the system be easily used by any interested person?	18	2	0

** Only 20 respondents were able to return the questionnaire*

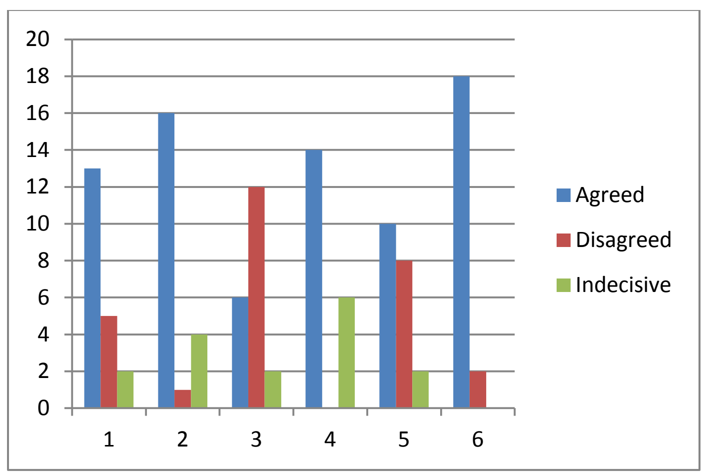

Fig. 16. Chart showing the performance of the System

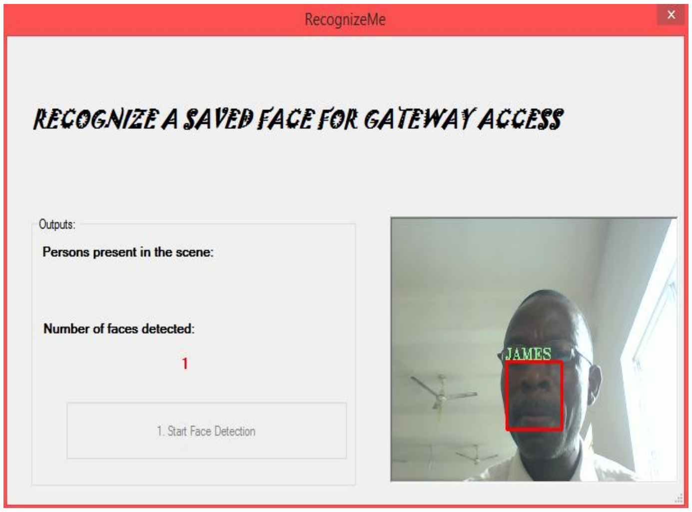

Fig. 17. An interface showing a recognized face

VII. CONCLUSION AND RECOMMENDATION

Conclusion

Having fully explored and implemented this new system using the genetic algorithm optimization for matching the features extracted from both pre and post images, the developed system was compared with some existing system such as, DWT (Discrete Wavelet Transform), PCA (Principal Component Analysis) and Biharmonic Equation and it out-performed the old system in terms of cost effectiveness, accuracy and simplicity of usage. In this work, it is clearly pointed out that the interface page where faces are captured gives room for the face to be saved with the username or userid as well as even pronouncing the name when it has been detected after it has been compared with numerous faces in the database which make it possible for the authenticity of the face to be searched for. During the process of capturing the faces, it was discovered that the illumination of the room or place where the capturing took place had a serious impact either negatively or positively on the effect of the process which eventually caused the process to be slow.

Recommendation

This study has proven that the Genetic Algorithm using multi-objective evolutionary granular computing can be used for face recognition even if the face is deformed either intentionally or accidentally. In the future, further research can be conducted to detect other algorithms that can also be implemented for face recognition which can address the factor of timing which this research could not cater for to get a better system that would perform accurately, effectively and timely. Also, when heavy make-up is applied to a face, researcher could make a frantic effort to see the effect on the system developed.

REFERENCES

[1]. A. Singh, Face Recognition: *Textbook on Biometrics*, India pp 26-39. 2012.

[2]. B. Heisele and T. Poggio, "Face recognition: Component-based versus global approaches," *Computer. Vis. Image Understand*., vol.91, 2003. pp. 6–21.

[3]. R.V. Singh, Plastic surgery: a new dimension to face recognition Information Forensics and Security, *IEEE Transactions* on 5, 2010. 441-448.

[4]. L. T. Young, *Image Processing Fundamentals Textbook* Version 2.3. 2007.

[5]. S. Gupta, Image Edge Detection: A Review, *International Journal of Advanced Research in Computer EngineringAnd Technology*, Vol 2, Issue 7. 2013.

[6]. L. Bin, Comparison for Image Edge Detection Algorithms, *IOSR Journal of Computer Engineering*, Vol 2, Issue 6, 2012.pp 01-04.

[7]. X. L. Xu,. Application of MATLAB in Digital Image Processing, *Modern Computer*, 2008. 43 (5), 35-37

[8]. B. Gupta, Image Denoising with Linear and Non-Linear Filters: A Review, *International Journal of Computer Science Issues*, 2013. Vol 10, Issue 6, No 2

[9]. I. K. Maita, S. Nag and S. K. Bandyopadhyay, A Novel Edge Detection Algorithm for Digital Mammogram, *International Journal of Information and Communication Technology Research*, 2012, Vol. 2. No 2.

[10]. D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. Journal of Computer.* Vis., vol. 60, no. 2, 2004. pp. 91–110.

[11]. R. Maini and H. Aggarwal, Study and Comparison of Various Image Edge Detection Techniques". *International Journal of Image Processing (IJIP)*, 2006. Vol. 3, Issue 1.