DOI: 10.21917/ijivp.2023.0427

GAUSSIAN DIFFUSIVE HARTIGAN MULTIDIMENSIONAL DEEP BELIEF DIVERGENCE FEATURE LEARNING USING PARTIAL DIFFERENTIAL EQUATION FOR FACE RECOGNITION

S. Moorthi and S. Karthikeyan
Department of Mathematics, Government Arts College, Salem, India

Abstract

Human face recognition has the most noteworthy role in detecting a person in many real-world scenarios in computer vision like identification, authentication, security, and so on. Face recognition typically acquires the features and compares them to a dataset to discover the best match. The existing methods failed to accurately extract the robust features for face recognition. To solve these issues, Gaussian Diffusive Hartigan Multidimensional Deep Belief Divergence Feature Learning (GDH-MDBDFL) method is proposed based on the Fractional Partial Differential Equation (FPDE) for face recognition. The proposed GDH-MDBDFL method is designed to improve the accuracy of face recognition using FPDE. The proposed GDH-MDBDFL method comprises three different layers such as input, output, and four hidden layers. Both qualitative and quantitative result analysis is presented to verify the effectiveness of the proposed GDH-MDBDFL method. The simulation results, the proposed GDH-MDBDFL method gives the higher recognition accuracy, and precision with lesser recognition time compared to the conventional methods.

Keywords:

Face Recognition, Feature Learning, Fractional Partial Differential Equation

1. INTRODUCTION

Human face recognition has the most noteworthy role in detecting a person in many real-world scenarios in computer vision like identification, authentication, security ,surveillance system, human-computer interaction, antiterrorism, psychology, and so on [1] - [3]. It is not an intrusive technique (i.e., not convey any health risks like the corona virus), and it does not need to touch anything during the acquisition level. Due to the complex and multidimensional structure of the face, it needs enormous processes and computation. On the other hand, face observation is a major designed visual perceptual capability in human beings. Face recognition is the task to learn one or more person in images. Face recognition typically acquires the features and compares them to a dataset to discover the best match. The facial feature extraction task is the initial stage for face recognition in the field of computer vision. But, the existing methods failed to accurately extract the robust features for face recognition. To solve these issues, fractional orders need to be calculated for improving accuracy. The field of fractional partial differential equations (FPDE) has drawn immense consideration towards theoretical [4],[5] and applied research [6]. In recent years, FPDE-based methods emerged to be powerful tools for image, signal, video, and optical fringe processing. They are used to enhance the quality of images in edge detection, segmentation, restoration, shape analysis, tracking, pattern, face, and action recognition. Therefore, a novel method based on the FPDE is necessary for improving face recognition.

Gaussian Diffusive Hartigan Multidimensional Deep Belief Divergence Feature Learning (GDH-MDBDFL) method is proposed based on the Fractional Partial Differential Equation (FPDE) for face recognition. The proposed GDH-MDBDFL method introduces a learning model based on fractional PDE for feature extraction to carry out face recognition. The proposed GDH-MDBDFL method is designed to improve the accuracy of face recognition using FPDE. The proposed GDH-MDBDFL method comprises three different layers such as input, output, and four hidden layers. At first, the number of face images is used as input in the input layer. In hidden layer one, regularized anisotropic diffusion filtering is applied to preprocess the images for eradicating the noise and preserving the images. Then, Hartigan's segmentation method is used in the next layer where the images are partitioned into similar groups depending on the pixel intensity. After that, FPDE is used to extract the diverse types of robust and pertinent features in the next layer. Lastly, feature learning is carried out using a multidimensional Bregman divergence classifier by computing divergence between training and testing features for face recognition. This in turn, the performance of face recognition is enhanced with higher accuracy and precision.

The major contributions of this paper include the following: 1) Construction of DGM-HLFL method via PDE for face recognition for accurate face recognition; 2) Regularized anisotropic diffusion filtering based preprocessing 3) Hartigan's Clustering based Segmentation Method 4) Fractional partial differential equations are used in the proposed GDH-MDBDFL method for pertinent feature extraction 5) The feature learning is performed by using the Multidimensional Bregman Divergence classifier in the proposed GDH-MDBDFL method for face recognition. 4) Experimental comparisons with other PDE-based methods demonstrated that the proposed GDH-MDBDFL method performs more significantly in terms of recognition accuracy, recognition time and false positive rate.

The structure of this document is as follows: s Section 2 discusses the background and the work related to our study, Section 3 describes the detailed methodology used for the proposed model, Further, Section 4 presents the analysis and visualization of the experimental results followed by the conclusion in Section

2. RELATED WORK

Over the past few years, deep neural networks which are made up of countless multiple nonlinear transformations have proven to be dominant. Based on PDEs and the wavelet transform, an image recognition algorithm was developed in [9]. The approach used in this work maintained the dominance of the second and fourth-order partial differential recognition while integrating high-order

PDE using weight coefficients. This improved the likelihood of maintaining images edge information, leading to better identification outcomes.

Specifically used in the analysis of speech, picture, and video data, deep convolutional neural networks (CNNs) were given a novel PDE interpretation in [10]. [11] looked at a theoretical analysis of deep neural networks and PDE for object recognition.

Deeply learnt features need to be both distinct and discerning in order to recognize faces. Label prediction in CNNs is not consistently appropriate because it is discovered that the deeply learned characteristics are inappropriate to aggregate all of the likely testing identities for training. For facial recognition, the new features are insufficiently effective.

Deep learning algorithms have been used in a variety of fields recently, such as image classification, face or image identification, visual tracking, and PDE solution, which have sped up the advancement of deep learning technology. In [11,12], a brand-new general model based on the Fourier periodic expansion function and the adaptive differential equation was put out. This form of expansion function also decreased the amount of time needed to compute for object recognition. Systolic Gaussian elimination and systolic Gauss-Jordan elimination, two separate functions, were combined in [13] to improve imperfect face recognition.

A framework for deep learning was created in [8, 14] to precisely recognize human behaviours. The features in video image sequences that are reduced in complexity were detected using the particle swarm optimization detection technique. It did not, however, concentrate on accelerating processing speed. For more accurate face recognition, Multi Task Cascaded Neural Network (MTCNN) and pre-trained FaceNet were introduced in [10]. The false positive rate, however, was not taken into account.

The above-mentioned research publications serve as inspiration for our study, which proposes GDH-MDBDFL via PDE for face recognition. The next sections contain the in-depth description.

3. PROPOSED METHODOLOGY

Feature learning is an imperative task in pattern recognition (such as image categorization). But, most feature-learning methods are not able to recognize relevant and irrelevant features. This limits the enhancement of classification results, mainly for large sample images. To solve this problem, Gaussian Diffusive Hartigan Multidimensional Deep Belief Divergence Feature Learning (GDH-MDBDFL) method based on Fractional Partial Differential Equation (FPDE) is proposed for accurate face recognition. In the GDH-MDBDFL method, feature selection is performed based on the FPDE which identifies the more robust features for recognition and eliminates irrelevant one to avoid complexity. In addition, feature learning is performed to classify the face images for recognition. The block diagram of the GDH-MDBDFL method is given in Fig.1.

The Fig.1 shows a block diagram for the overall process involved in the proposed GDH-MDBDFL method to accurately recognize the faces images using four different steps. First, the number of face images is X_1, X_2, X_3,…..X_n are gathered from the Extended Yale B dataset and Pie dataset. Then the regularized anisotropic diffusion filtering is used to perform image preprocessing. After that, Hartigan's clustering segmentation method is employed to segment the images into similar regions. Then, the robust feature extraction is performed by using FPDE. Lastly, a multidimensional Bregman divergence classifier is employed to learn the features of face recognition.

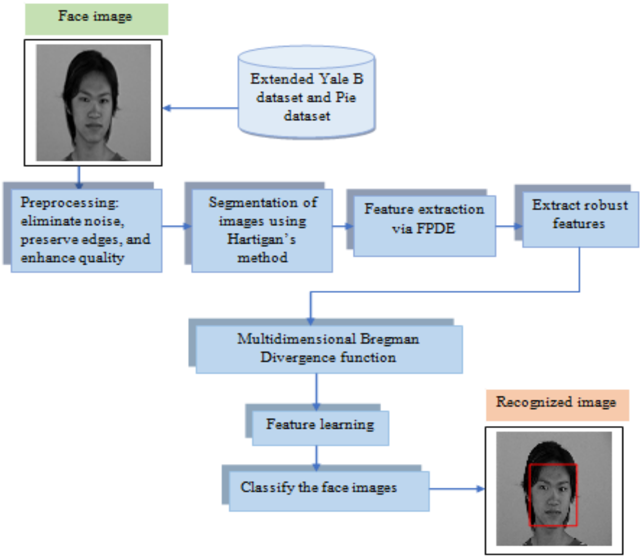


Fig.1. Block Diagram of GDH-MDBDFL using FPDE for Face Recognition

The proposed GDH-MDBDFL method uses the deep belief neural network model. It comprises several layers such as the input layer, output layer, and four hidden units i.e. layers. The input and output layers are a visible unit. The structure of the deep belief neural network is demonstrated in Fig.2.

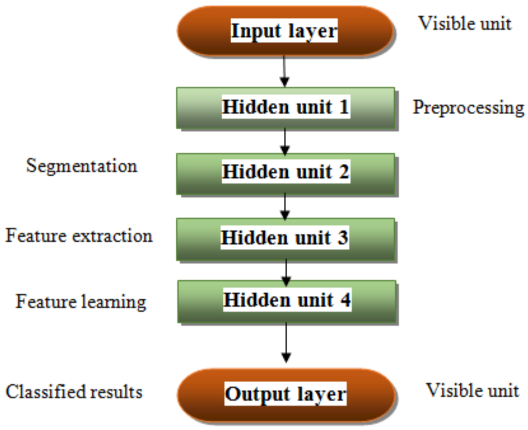


Fig.2. Structure of Deep Belief Neural Network

The Fig.2 illustrates the structure of a deep belief neural network for face recognition. The designed deep belief neural network comprises four hidden units and two visible units. Each unit in the network includes the number of neurons (i.e. nodes) that are associated from one layer to another layer to create the entire network. The proposed deep belief network is worked in a feed-forward manner. It means the input is forwarded from the previous unit into the next consecutive unit. The input face image is forwarded to perform preprocessing, segmentation, feature extraction, and classification process for recognition. The above-mentioned processes are explained as follows.

3.1 REGULARIZED ANISOTROPIC DIFFUSION FILTERING BASED PREPROCESSING

In the proposed GDH-MDBDFL method, the preprocessing is initially carried out to remove the noise presented in the image for face recognition. In the proposed GDH-MDBDFL method, the face images from the input layer are given to preprocessing process in which the noise in the input image is eliminated. The preprocessing is performed by using regularized anisotropic diffusion filtering. The Regularized anisotropic diffusion filtering model is employed for eradicating the noise pixels from the input image. The process of regularized anisotropic diffusion filtering is illustrated below Fig.3.

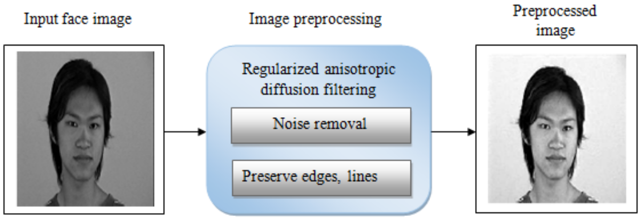


Fig.3. Regularized Anisotropic Diffusion Filtering based Preprocessing

The Fig.3 depicts the overall process of face image preprocessing using regularized anisotropic diffusion filtering. By applying the filtering model, a noteworthy part of the image information such as edges, and lines are not affected for further image analysis. Several existing methods were employed for preprocessing however, the image quality was not improved by preserving the edges. Consider input face image $u_0(x,y)$ and the pixels are denoted by a_1, a_2,a_3,…,a_n. These face images are placed in a filtering window through the size of 3*3 in the form of rows and columns. The pixels are organized in a filtering window in increasing order using regularized anisotropic diffusion filtering technique. Here, the center value is considered from the filtering window. The center of the filtering window has even numbers of pixels, the average of these two pixels is taken as the center value.

To get the preprocessed image, anisotropic diffusion filtering is used and it is mathematically formulated as follows:

$$\frac{\partial u(x,y,t)}{\partial t} = -div\left[a\left(\left|\nabla u(x,y,t)\right|\right)\nabla u(x,y,t)\right] \tag{1}$$

$$\frac{\partial u(x,y,t)}{\partial n} = 0,\ \partial\Omega \times (0,T] \tag{2}$$

$$u(x,y,0) = u_0(x,y) \tag{3}$$

where, $u(x,y,0)$ refers the original image intensity function, $u(x,y,t)$ refers the smoothed image intensity values in the position $(x,y) \in \Omega R^2$ at time t, ∇ refers the gradient, $\partial\Omega$ is the smooth boundary, $u_0(x,y)$ refers a original image, div is a divergence operator, $\frac{\partial u(x,y,t)}{\partial n}$ refers the derivative in normal direction to the boundary, and $a(|\nabla u(x,y,t)|)$ refers the diffusion coefficient function to preserve edges in the input image, also called edge-stopping function. The edge-stopping functions lead to backward-forward difficulties. To solve this issue, regularized version of anisotropic diffusion filtering convolved through a Gaussian within the non-linearity and it is mathematically formulated as follows.

$$\frac{\partial u(x,y,t)}{\partial t} = -div\left[a\left(\left|\nabla G_\sigma * u(x,y,t)\right|\right)\nabla u(x,y,t)\right] \tag{4}$$

where, Ω refers a bounded domain of R^2 with an appropriately smooth boundary, n is a unit outer normal to Ω and T>0. The term $\nabla G_\sigma \times u(x,y,t)$ is the regularized version of $(\nabla u(x,y,t))$ convolved with Gaussian distribution at time t.

To achieve a good trade-off between noise removal and edge preservation, fractional order partial differential equation is used in the preprocessing process.

$$\frac{\partial u(x,y,t)}{\partial t} = -div^\alpha\left[a\left(\left|\nabla^\alpha G_\sigma * u(x,y,t)\right|\right)\nabla^\alpha u(x,y,t)\right] \tag{5}$$

where, the fractional order is $\alpha(0\leq\alpha\leq 2)$ and $u(x,y,t)$ refers the smooth gray scale image at time t. The fractional-order gradient vector with α order is described as,

$$\nabla^\alpha u(x,y,t) = \left[G_\sigma \nabla_x^\alpha u(x,y,t), \nabla_x^\alpha u(x,y,t)\right] \tag{6}$$

where, $\nabla_x^\alpha u(x,y,t)$ refers the partial-order derivative of $u(x,y,t)$ with respect to the variable x whose order is α. Subsequently, Gaussian function is measured as follows.

$$G_\sigma = \exp\left[-\frac{1}{2} * \frac{\|a_c - a_n\|^2}{\sigma^2}\right] \tag{7}$$

where, G_d specifies a Gaussian distribution, a_c specifies the center pixel in the filtering window, a_n specifies the pixel in the filtering window, and d specifies the deviation. By analyzing the above equations, the pixels varied from the center vale and are known as noisy pixels. The noisy pixels introduce the strong diffusion action and it is eradicated and the image contrast is improved. With this, quality-enhanced images are acquired to improve the face recognition process. The pseudo-code representation of regularized anisotropic diffusion filtering based preprocessing is given as below.

Algorithm 1 Regularized Anisotropic Diffusion Filtering based Preprocessing

Input: Dataset D, Face Images
Output: Preprocessed face images

Step 1: Begin
Step 2: For each face image
Step 3: Apply anisotropic diffusion filtering

$$\frac{\partial u(x,y,t)}{\partial t} = -div\left[a\left(\left|\nabla u(x,y,t)\right|\right)\nabla u(x,y,t)\right]$$

Step 4: Derive regularized version of anisotropic diffusion filtering convolved through a Gaussian function

$$\frac{\partial u(x,y,t)}{\partial t} = -div\left[a\left(\left|\nabla G_\sigma * u(x,y,t)\right|\right)\nabla u(x,y,t)\right]$$

Step 5: Use fractional order partial differential equation for achieving a good trade-off between noise removal and edge preservation,

$$\frac{\partial u(x,y,t)}{\partial t} = -div^{\alpha}\left[a\left(\left|\nabla^{\alpha}G_{\sigma} * u(x,y,t)\right|\right)\nabla^{\alpha}u(x,y,t)\right]$$

Step 6: Compute fractional-order gradient vector with α order

$$\nabla^{\alpha}u(x,y,t) = \left[G_{\sigma}\nabla_x^{\alpha}u(x,y,t), \nabla_x^{\alpha}u(x,y,t)\right]$$

Step 7: Measure Gaussian function

$$G_{\sigma} = \exp\left[-\frac{1}{2} * \frac{\|a_c - a_n\|^2}{\sigma^2}\right]$$

Step 8: Remove noisy pixels
Step 9: Preserve edges, lines and etc.
Step 10: Increase the image contrast
Step 11: Return (preprocessed image)
Step 12: End for
Step 13: End

Algorithm 1 shows the step-by-step process of preprocessing based on regularized anisotropic diffusion filtering. By applying the designed preprocessing model, the noise in the image is eliminated to get the contrast-enhanced images.

3.2 HARTIGAN'S CLUSTERING BASED SEGMENTATION METHOD

Segmentation of face images is performed in the next hidden layer. Face image segmentation is the procedure of splitting the input images into similar regions. It provides favorable information in image processing. But, it is a difficult task due to the uneven form and confusing boundaries of face images. In the past, various works have been planned for the segmentation of face images. But, the accuracy was not improved enough. Therefore, Hartigan's clustering method is employed to cluster the images into several regions with better accuracy.

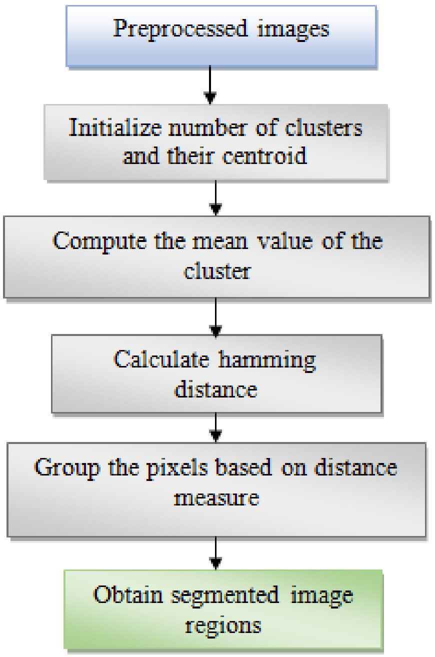


Fig.4. Hartigan's Clustering based Segmentation Method

The Fig.4 shows the process involved in Hartigan's face image segmentation method. Hartigan's segmentation method takes preprocessed images as input where the images are divided into diverse region depended on the pixel similarity. Hartigan's segmentation method partitions the image regions into groups or clusters regarding the nearest mean.

Let consider the number of pixels in an image $p_1, p_2, p_3, \ldots p_n$. At first, Hartigan's segmentation method initializes the number of clusters and their centroid (mean) randomly.

$$\beta_i = \beta_1, \beta_2, \beta_3, \ldots \beta_q \tag{8}$$

$$\gamma_i = \gamma_1, \gamma_2, \gamma_3, \ldots \gamma_q \tag{9}$$

where β_i refers the number of clusters and γ_i refers the number of clusters centroid. For each initialized cluster, mean value is measured as given below,

$$M = \frac{\sum_{i=1}^{n} p_i}{n} \tag{10}$$

where M refers a mean of the particular cluster and it is computed as the ratio of the sum of all the pixels p_i in the particular cluster to the total number of pixels n. Then, the Hartigan's segmentation technique partitions the pixels into different clusters based on the distance between pixels.

In the proposed segmentation process, hamming distance is determined to find the deviations between pixels and their mean value in the cluster.

$$H_m = \sum_{i=1}^{n} |\mu - p_i| \tag{11}$$

where, H_m denotes the hamming distance, μ denotes the mean value and p_i point outs the preprocessed image pixels. The clustering process reduces the average distance from the mean within the cluster. With this, the pixel closer to the mean or minimum distance is clustered into the certain cluster.

$$Y = \arg\min H_m \tag{12}$$

where, arg min refers an argument of the minimum function. With this, the minimal distance among pixels and their mean value is considered for segmenting the images into similar regions. The pseudo-code representation of Hartigan's segmentation method is given as below.

Algorithm 2: Hartigan's Clustering based Segmentation Method

Input: Preprocessed images
Output: Image segmentation

Step 1:Begin
Step 2: **For** each preprocessed image
Step 3: Initializes the number of clusters

$$\beta_i = \beta_1, \beta_2, \beta_3, \ldots \beta_q$$

Step 4: Initializes number of centroid

$$\gamma_i = \gamma_1, \gamma_2, \gamma_3, \ldots \gamma_q$$

Step 5: Compute mean value for each cluster

$$M = \frac{\sum_{i=1}^{n} p_i}{n}$$

Step 6: Calculate hamming distance to determine the deviations between pixels and their mean value in the cluster

$$H_m = \sum_{i=1}^{n} |\mu - p_i|$$

Step 7: Pixel closer to the mean is clustered into the certain cluster

$$Y = \arg\min H_m$$

Step 8: Perform segmentation
Step 9: End for
Step 10: End

Algorithm 2 describes the process involved in Hartigan's clustering based segmentation method where the similarity between pixels is computed to partition the images into similar regions. The pixel with higher similarity is segmented in Hartigan's method.

3.3 FEATURE EXTRACTION

Once the segmentation is performed, the feature extraction process is carried out in the third hidden layer. Feature extraction is an important step in pattern recognition (such as image classification). However, most feature extraction methods are not able to extract robust features. This limits the improvement of recognition (classification) results, especially for huge sample sizes. To solve this problem, fractional partial differential equations are used in the proposed GDH-MDBDFL method for pertinent feature extraction. The segmented face images are given as input to the fractional partial differential equations. The extracted pertinent features of the face images are the FPDE's output.

Firstly, it is considered that the image processing task to be done can be defined through evolution equations, i.e., the input image evolves based on a specified evolution equation, and the result of evolution is the desired processing node. So the first thing is to build a unified intelligent system of partial differential equations to describe evolution. The established fractional partial differential equation system is composed of two-coupled FPDE. The evolution function of output extracted features is relevant and irrelevant. The pertinent one is used for the recognition process and the irrelevant one is removed. Thus, the two coupled FPDEs is given by,

$$\frac{\partial u(x,y)}{\partial y} = R(x,t)\frac{\partial^{\alpha} u(x,y)}{\partial_R x^{\alpha}} + IR(x,t)\frac{\partial^{\alpha} u(x,y)}{\partial_{IR} x^{\alpha}} + s(x,t) \quad (13)$$

where, ∂ is considered as $1\leq\partial\leq2$, $s(x,t)$ is the two face images, the function $R(x,t)$ and $IR(x,t)$ are denoted as relevant and irrelevant features (less informative features for recognition).

When $\alpha=2$ and setting $c(x,t) = R(x,t) + IR(x,t)$, then the above equation becomes,

$$\frac{\partial u(x,y)}{\partial t} = C(x,t)\frac{\partial^2 u(x,y)}{\partial x^2} + s(x,t) \quad (14)$$

When $\alpha=1$ and setting $c(x,t) = R(x,t) + IR(x,t)$, then the above equation becomes,

$$\frac{\partial u(x,y)}{\partial t} = C(x,t)\frac{\partial u(x,y)}{\partial x} + s(x,t) \quad (15)$$

By using the above Eq.(15), the robust features from the face images are extracted. The irrelevant features are removed to minimize the time complexity involved in the recognition process. By applying FPDEs for feature extraction, 12 features are extracted into various types of features such as geometric, color, texture, spatial, and fractal and face image-oriented features. Several studies associated with face recognition use texture features to get better classification results. The proposed GDH-MDBDFL method combines the five types of features such as geometric, color, texture, spatial, and fractal to increase the accuracy of face recognition.

Spatial features are the features extracted and summarized directly from grid information. It implicitly contains spatial relations among semantically important parts of the image. Examples of spatial features are edges, contrasts, and a set of intensity statistics. Besides, geometric features are a set of features of an image composed of geometric components like points, lines, curves, or surfaces. The texture features are contrast, correlation, entropy, homogeneity, and so on. The mean, standard deviation, and skewness of an image are color referred to as the color moments employed to extract the features. Fractal features are considered to compute self-similarity. The extracted features from the input segmented image are Eccentricity, Extent, Energy, Entropy, Homogeneity, Variance, Mean, Standard deviation, Skewness, Kurtosis, contrast, and correlation. These extracted features are given as follows.

3.3.1 Contrast:

The gray level intensity contrast of the image is measured as the difference between the pixel (a_i) and their neighboring pixels (a_j) in the set of the pixel.

$$\text{Contrast} = \sum_i\sum_j|p_i-p_j|^2 \quad (16)$$

where, p_i is the pixel and p_j is the neighboring pixel.

3.3.2 Correlation:

The correlation feature of the image gives the association of pixels intensities. It is computed by,

$$Correlation = \frac{\sum_i\sum_j (p_i-\mu_i)(p_j-\mu_j)}{\sigma_i * \sigma_j} \quad (17)$$

where μ_j and μ_j is a mean of the pixels p_i, p_j and σ_i and σ_j are the deviations.

3.3.3 Mean:

It refers as a standard intensity value of the pixel.

$$\mu = \frac{1}{N_p}\sum_{r=1}^{m}\sum_{c=1}^{n} k(r,c) \quad (18)$$

where, N_p is the number of pixels of an image, and $k(r,c)$ is the value of the equivalent pixel at row r and column c correspondingly.

3.3.4 Standard Deviation:

This feature point outs the variation among the pixels obtainable over an input image.

$$\sigma = \sqrt{\frac{1}{N_p}\sum_{r=1}^{m}\sum_{c=1}^{n}\left(k(r,c)-\mu\right)^2} \quad (19)$$

3.3.5 Entropy:

It computes the input image's texture volatility the value is high when increasing the randomness.

$$Entropy = -\sum_{i,j} g(i,j)\log_2 g(i,j) \quad (20)$$

3.3.6 Variance:

The dependent linear gray position between pixels in a certain positions associated to others is referred as variance.

$$v^2 = \sum_{i=0}^{L-1}\left(p\left(Z_i\right)-\mu\right)^2 \tag{21}$$

where v^2 specifies a variance, $p(Z_i)$ denotes an intensity of pixel, μ symbolizes a mean.

3.3.7 Kurtosis:

It is a measure of probability distribution with a peak in the real-valued random variable.

$$\delta = \frac{1}{\left(N_p-1\right)\sigma^4}\sum_{r=1}^{m}\sum_{c=1}^{n}\left(k\left[r,c\right]-\mu\right)^4 \tag{22}$$

where, δ denotes the Kurtosis.

3.3.8 Skewness:

It is computed as the probability distribution by asymmetry present in the real-valued random variable.

$$\vartheta = \frac{1}{\left(N_p-1\right)\sigma^3}\sum_{r=1}^{m}\sum_{c=1}^{n}\left(k\left[r,c\right]-\mu\right)^3 \tag{23}$$

3.3.9 Eccentricity:

An eccentricity is a non-negative real value to defines its image or figure separately. The output of eccentricity is ranged between 0 to 1. It is computed by,

$$Eccentricity = axislength_{short} / axislength_{long} \tag{24}$$

3.3.10 Extent:

It is described as the region of the image object partitioned through the rectangle area that limits the size of the objects.

$$Extent = (net\ area)/(bounding\ rectangle) \tag{25}$$

3.3.11 Energy:

Energy of the pixels is defined as a count of repeated pixels. It computed by,

$$Energy = \sum_{i,j} g(i,j)^2 \tag{26}$$

where $g(i,j)$ is a frequency value at the coordinates i and j.

3.3.12 Homogeneity:

It is an estimation of proximity of the homogenization feature to the distribution.

$$Homogeneity = \sum_{i,j}\frac{g\left(i,j\right)}{1+\left(i-j\right)^2} \tag{27}$$

3.4 MULTIDIMENSIONAL BREGMAN DIVERGENCE CLASSIFICATION

After the feature extraction, feature learning is carried out at the next hidden layer for classifying the face images. The feature learning is performed by using the Multidimensional Bregman Divergence classifier in the proposed GDH-MDBDFL method for face recognition. A multidimensional Bregman divergence classifier is a machine learning technique for learning the given input of extracted features. The Bregman divergence classifier uses the geometric, color, texture, spatial, and fractal features to recognize the face images. Thus, the name is called a multidimensional Bregman divergence classifier.

In the proposed GDH-MDBDFL method, Bregman Divergence Function computes the similarity between two extracted features (i.e., training face image features and testing features). The extracted features are first arranged into the matrix in the form of rows and columns.

$$A = \begin{bmatrix} q_{11} & q_{12} & \cdots & q_{1n} \\ q_{21} & q_{22} & \cdots & q_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ q_{n1} & q_{n2} & \cdots & q_{nn} \end{bmatrix} \tag{28}$$

Accordingly, the Bregman divergence function is mathematically measured as follows,

$$b_F = F(q_i)-F(q_j)-\langle\nabla F(q_j),q_i- q_j\rangle \tag{29}$$

where, b_F refers to the Bregman divergence function, q_i refers to the testing features and q_j a training feature. The similarity between the features is measured based on the divergence between features. The lesser divergence between features than the threshold is classified as the best match in face recognition. The feature with higher divergence is classified as not matched in face recognition.

Subsequently, the error rate is computed for each classified result and it is given by,

$$E_r = A_b - P_b \tag{30}$$

where, E_r is the error rate and it is measured based on the difference between actual classified results A_b and predicted classified results P_b. Depending on the error rate the weights are updated and determine the minimum error output with the aid of the gradient descent function as provided below.

$$w' = \eta\frac{\partial \log b_F}{\partial w_{(t)}} + w\left(t\right) \tag{31}$$

where, w' is an updated weight, $w(t)$ is a current weight, η is a learning rate and $\frac{\partial \log b_F}{\partial w_{(t)}}$ is a gradient descent function i.e., a first-order iterative algorithm to determine a local minimum of a differentiable function with respect to error. From that, accurate classification is achieved to improve face recognition performance. The pseudo-code representation of regularized anisotropic diffusion filtering based preprocessing is given as below. The pseudo code for multidimensional Bregman divergence classifier is given provided as follows.

Algorithm 3 Multidimensional Bregman Divergence Classifier

Input: Extracted features **Output:** Face recognition
Step 1: **Begin** **Step 2**: **For** each extracted feature q_j Step 2: Perform feature leaning using Bregman divergence function **Step 3:** Compute multidimensional Bregman divergence function $b_F = F(q_i)-F(q_j)-\langle\nabla F(q_j),q_i- q_j\rangle$ **Step 4**: **If** $b_F<$ *threshold*, **then** **Step 5:** Match in face recognition

Step 6: **Else**
Step 7: Not match in face recognition
Step 8: Calculate the error

$$E_r = A_b - P_b$$

Step 9: Updated the weight using gradient descent function

$$w' = \eta \frac{\partial \log b_F}{\partial w_{(t)}} + w(t)$$

Step 10: Obtain minimum error classification result
Step 11: return (face recognition)
Step 12: **End if**
Step 13: **End for**
Step 14: **End**

Algorithm 3 describes the process of a Multidimensional Bregman divergence classifier for face recognition. Multidimensional Bregman divergence function is applied for each extracted feature where the similarity between features (training and testing features) is computed to classify the images. In addition, error rate and weight updation are carried out to get the accurate classification results of face recognition.

4. EXPERIMENTAL RESULTS

Experimentation of the proposed GDH-MDBDFL method based on factional PDE for face recognition is implemented using MATLAB version 15b. The experiment is performed with the system specifications of a core i5 processor with 233 MHz, 8GB RAM, and 1 TB hard disk. The results of face recognition are analyzed by using the Extended Yale B dataset and Pie dataset. The extended Yale Face Database B is obtained from https://www.kaggle.com/ tbourton/ extyaleb croppe dpng?select=CroppedYalePNG. The extended Yale Face Database B includes 16128 images obtained from 28 distinct human subjects under 9 poses and 64 illumination conditions. All test image data used in the organized and manually cropped. Lastly, the images are re-sized to 168×192 images. The data format is similar to the Yale Face Database B. Pie dataset is extracted from http://robotics.csie.ncku.edu.tw/ Databases/FaceDetect_Pose Estimate.htm#Our_Database. The Pie dataset comprises of 6660 images obtained from 90 distinct subjects. Here, each subject includes 74 images. Among the 74 images, 37 images are obtained every 5 degrees from right profile, referred to as +90°. The remaining 37 images are acquired every 5 degrees from left profile, denoted to as -90° respectively in the pan rotation. The first part of the image is ‘A’ and ‘B’ represents the real-shot or synthesized image correspondingly. The two digit number is denoted as a subject number. The rest of the filename represents the equivalent facial pose. To conduct the experimental assessment, ten iterations are performed with diverse input face images in the ranges of 1500 to 15000 for Extended Yale B dataset and 600 to 6000 for Pie dataset. The obtained result of the GDH-MDBDFL method is compared with existing Fractional-Order (FO) Lorenz chaotic system for cancellable biometric recognition scheme (FO Lorenz chaotic system) [1] and Fractional-order Chebyshev Moment Invariants (FCMI) [2]. Diverse testing metrics are used for analyzing the performance of proposed and existing methods. At first, a qualitative analysis of the GDH-MDBDFL method is performed. Followed by, quantitative analysis of the proposed GDH-MDBDFL method and existing FO Lorenz chaotic system and FCMI methods are discussed with different parameters such as recognition accuracy, recognition time, and precision based on the number of face images.

Recognition accuracy is defined as the ratio of a number of face images accurately recognized to the total face images involved in the simulation. Recognition accuracy is measured in the unit of percentage (%). Recognition time is measured as the time needed by the algorithm to recognize the face. It is computed in terms of milliseconds (ms). Precision is defined as the ratio of predicting positive samples in the face image dataset. Precision is computed in terms of percentage (%).

$$R_{acc} = \sum_{i=1}^{n} \frac{FI_{AR}}{Fl_i} \times 100 \tag{40}$$

$$R_{time} = \sum_{i=1}^{n} Fl_i \times Time[FR] \tag{41}$$

$$Precision = TP/(TP+FP) \times 100 \tag{42}$$

where, R_{acc} is the recognition accuracy, FI_{AR} is the number of face images accurately recognized and FI_i is the face images used in the simulation process. R_{time} refers a recognition time, *Time* [FR] refers a time consumed in the face recognition process, *TP* denotes the True Positive (face images correctly classified as matched) and *FP* denotes the False Positive (face images incorrectly classified as not matched).

The qualitative analysis of the proposed GDH-MDBDFL method for the Extended Yale B dataset and Pie dataset is discussed with various processes such as preprocessing, segmentation, feature extraction, and classification.

Fig.5(a). Input Face Images from Pie Dataset

Fig.5(b). Input Face Images from Extended Yale Face B dataset

Fig.6(a). Preprocessed images from Pie dataset

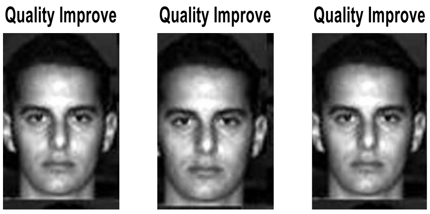


Fig.6(b). Input Face Images from Extended Yale Face B dataset

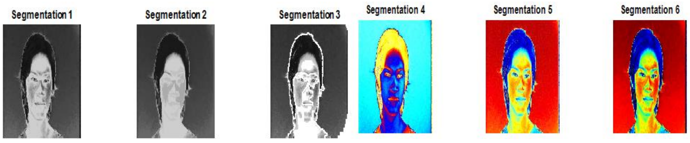


Fig.7 (a) Segmented Images from Pie Dataset

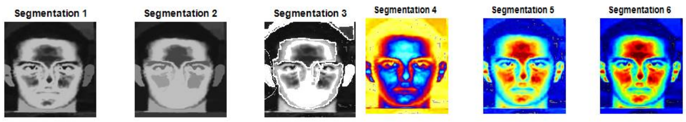
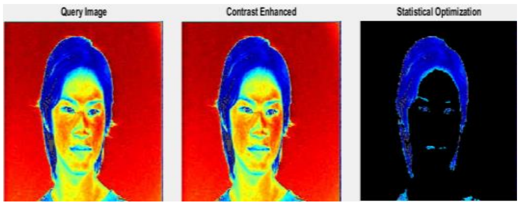
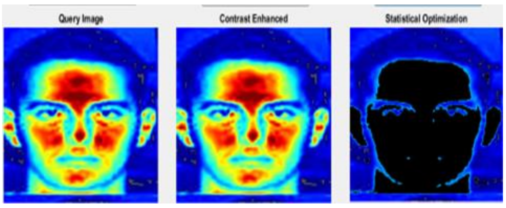


Fig.7(b). Input Face Images from Extended Yale Face B Dataset



(a)



(b)

Fig.8(a). Feature extracted images from Pie dataset (b) Input face images from Extended Yale Face B dataset

(a)

(b)

Fig.9(a). Face recognition images from Pie dataset (b) Input face images from Extended Yale Face B dataset

The Fig.5 to Fig.9 show the qualitative analysis of the GDH-MDBDFL method for the Pie dataset and Extended Yale B dataset. At first, the input face images are collected from the given dataset, and preprocessing images are obtained using regularized anisotropic diffusion filtering. Then the segmentation of different regions is acquired to decrease the time consumption of face recognition. Then the FPDE extracts the features. Finally, the face is correctly recognized with the extracted features. From the above qualitative analysis, the proposed GDH-MDBDFL method provides efficient face recognition results.

The Table.1 describes the simulation results of recognition accuracy based on the number of face images collected from Extended Yale Face Database B dataset and Pie dataset. In the simulation process, the number of face images is considered from 1500 to 15000 from the Extended Yale Face Database B dataset and 600 to 6000 images are considered from Pie dataset. The obtained results of face recognition accuracy using proposed GDH-MDBDFL method are compared with the existing FO Lorenz chaotic system and FCMI. Ten iterations are performed with various input images. By observing the above table, recognition accuracy is improved in proposed GDH-MDBDFL method than the existing FO Lorenz chaotic system and FCMI in both the datasets. The graphical comparison of recognition accuracy is depicted in the following figure.

Table.1. Recognition accuracy (%)

(a) Extended Yale Face Database B dataset			
Face images	**Existing FCMI**	**Existing FO Lorenz chaotic system**	**Proposed GDH-MDBDFL**
1500	82.00	87.00	93.00
3000	81.23	86.82	92.25
4500	80.51	86.62	92.04
6000	79.63	85.14	91.86
7500	78.63	84.62	91.24
9000	77.52	83.62	90.57
10500	76.41	82.14	90.16
12000	75.63	80.35	89.52
13500	73.21	78.62	89.04
15000	72.42	77.14	88.63
(b) Pie dataset			
Face images	**Existing FCMI**	**Existing FO Lorenz chaotic system**	**Proposed GDH-MDBDFL**
600	84.00	89.00	96.00
1200	82.61	88.51	95.64
1800	82.42	88.05	95.15
2400	80.92	87.14	94.89
3000	80.63	86.31	93.68
3600	79.46	85.14	93.41
4200	78.82	84.21	92.95
4800	77.63	83.66	92.63
5400	76.14	82.04	91.24
6000	74.47	80.54	90.64

The Table.1 represent the face recognition accuracy of proposed GDH-MDBDFL method, existing FO Lorenz chaotic system and FCMI using Extended Yale Face Database B dataset and Pie dataset respectively. As represented in the above figures, number of face images is used as input in the x-axis and the accuracy for the three methods is obtained at y-axis. From the above graph, the face recognition accuracy of GDH-MDBDFL method is found to be higher than the other methods. For instance, with the input of 1500 face images, 1395 images correctly recognized in the GDH-MDBDFL method whereas existing FO Lorenz chaotic system and FCMI correctly recognized 1305 and 1230 images and the overall accuracy is obtained as 93%, 87% and 82% for Extended Yale Face Database B dataset. Similarly with the input of 600 face images, 576, 534 and 504 face images are correctly recognized in GDH-MDBDFL, FO Lorenz chaotic system and FCMI and accuracy is obtained as 96%, 89% and 84% respectively for Pie dataset.

The accuracy improvement in GDH-MDBDFL method is attained by applying multidimensional Bregman divergence classifier for face recognition. By applying the designed classifier, the extracted features are learned to recognize the face images. This can be done by measuring the divergence between testing features and training features. The feature with minimum divergence is classified as matched for face recognition. With this, face recognition accuracy is improved in proposed GDH-MDBDFL method by 9% and 17% as compared to FO Lorenz chaotic system and FCMI for Extended Yale Face Database B dataset. Also, face recognition accuracy of proposed GDH-MDBDFL method is increased by 11% and 19% as compared to FO Lorenz chaotic system and FCMI for Pie dataset.

Table.2. Recognition time using GDH-MDBDFL, FO Lorenz chaotic system and FCMI

(a) Extended Yale Face Database B dataset			
Face images	**Existing FCMI**	**Existing FO Lorenz chaotic system**	**Proposed GDH-MDBDFL**
1500	240.63	225.62	200.25
3000	275.61	260.48	245.62
4500	325.54	310.63	285.95
6000	355.17	340.52	320.56
7500	390.54	375.14	360.45
9000	435.63	420.15	400.42
10500	465.25	450.55	429.62
12000	485.14	470.52	455.48
13500	498.64	485.62	470.52
15000	515.25	500.84	480.75
(b) Pie dataset			
Face images	**Existing FCMI**	**Existing FO Lorenz chaotic system**	**Proposed GDH-MDBDFL**
600	200.2	180.25	160.52
1200	220.51	200.62	180.65
1800	225.48	210.63	195.63
2400	270.63	255.63	240.48
3000	295.14	280.78	265.82
3600	320.44	305.85	290.62
4200	360.95	345.24	330.54
4800	410.25	395.12	380.25
5400	428.52	410.33	410.48
6000	442.62	425.18	435.62

The Table.2 demonstrates the result analysis of recognition time for three methods such as GDH-MDBDFL method, existing FO Lorenz chaotic system and FCMI using Extended Yale Face Database B dataset and Pie dataset. The validation process is done by comparing the performance of proposed GDH-MDBDFL method with conventional algorithms. With the varying number of face images, different recognition time involved in both proposed and existing methods is measured. By analyzing the Table.2, time taken to recognize the face images is found to be lower in proposed GDH-MDBDFL method than the existing FO Lorenz chaotic system and FCMI. Graphical representation of recognition time for three methods is illustrated as follows.

The Table.2 show the recognition time results of proposed and existing methods based on the number of leaf images from Extended Yale Face Database B dataset and Pie dataset respectively. As demonstrated in the above figures, recognition time for three different techniques is increased with the increase in a number of face images. But, comparatively GDH-MDBDFL method uses a minimal time for recognition than the existing FO Lorenz chaotic system and FCMI. By considering 1500 face images in Extended Yale Face Database B dataset, recognition time is measured as 200.25ms, 2262 ms and 240.63 ms for GDH-MDBDFL method, FO Lorenz chaotic system and FCMI respectively. Similarly, in 600 face images using Pie dataset, recognition time is computed 160.52 ms, 180.25 ms, and 200.20 ms for the GDH-MDBDFL method, FO Lorenz chaotic system and FCMI respectively. From the comparison, GDH-MDBDFL provides minimal recognition time in all the runs.

On the contrary to existing methods, proposed GDH-MDBDFL method uses the FPDE to extract the robust features. The designed FPDE not only chooses the robust features, but it also finds and eliminates the irrelevant features for face recognition. It also extracts diverse types of features such as geometric, color, texture, spatial, and fractal features. These extracted features are more informative to recognize the face images with less time. Therefore, the recognition time of the proposed GDH-MDBDFL method is reduced by 5% and 9% compared to FO Lorenz chaotic system and FCMI for Extended Yale Face Database B dataset. In addition, the recognition time of the proposed GDH-MDBDFL method is decreased by 5% and 10% compared to FO Lorenz chaotic system and FCMI for Pie dataset.

Table.3. Precision using GDH-MDBDFL, FO Lorenz chaotic system and FCMI

(a) Extended Yale Face Database B dataset			
Face images	**Existing FCMI**	**Existing FO Lorenz chaotic system**	**Proposed GDH-MDBDFL**
1500	82.14	84.65	92.85
3000	82.04	84.31	92.04
4500	81.74	83.74	91.85

6000	81.25	82.61	91.22
7500	80.69	81.97	90.65
9000	79.63	81.52	89.63
10500	78.54	80.63	88.25
12000	77.25	79.14	87.62
13500	76.24	78.53	86.44
15000	75.14	77.11	85.17
(b) Pie dataset			
Face images	**Existing FCMI**	**Existing FO Lorenz chaotic system**	**Proposed GDH-MDBDFL**
600	86.51	88.52	96.45
1200	82.62	84.62	96.25
1800	80.47	82.54	95.97
2400	79.63	81.97	95.78
3000	77.14	80.56	95.41
3600	76.87	79.65	94.86
4200	75.63	78.41	94.72
4800	74.62	77.36	94.65
5400	73.17	76.52	93.88
6000	72.14	74.14	93.64

The Table.3 illustrates the experimental results of precision for GDH-MDBDFL method, FO Lorenz chaotic system and FCMI for Extended Yale Face Database B dataset and Pie dataset. In the simulation conduction, diverse ranges of face images are used as input. From the above table, performance of precision using GDH-MDBDFL method, FO Lorenz chaotic system and FCMI are analyzed and compared. The comparative analysis illustrates that the precision of the proposed GDH-MDBDFL method is improved than the existing FO Lorenz chaotic system and FCMI. According to the values in the above table, the graph for precision is plotted as given below.

The Table.3 shows the comparative graph of precision using proposed and existing methods for Extended Yale Face Database B dataset and Pie dataset respectively. A number of face images are taken in the horizontal axis and the results of precision are obtained at the vertical axis. To carry out a fair comparison between the methods, sample face images are used in the ranges of 1500-15000 for Extended Yale Face Database B dataset and 600-6000 for Pie dataset. With the input of 1500 plant leaf images, precision is obtained as 92.85%, 84.65% and 82.14% in GDH-MDBDFL method, FO Lorenz chaotic system and FCMI respectively. Also, 96.45%, 88.52% and 86.51% of precision is obtained in three methods for Pie dataset. The above discussion shows that the proposed GDH-MDBDFL method attained higher precision than the other methods.

The reason behind the improvement is due to the application of regularized anisotropic diffusion filtering, Hartigan's clustering segmentation method, and robust feature extraction. First, filtering technique is applied to remove the noise and improve the image contrast. These images are segmented through the clustering concept. With this, the pertinent features are acquired using FPDE for face recognition. As a result, precision is improved in the GDH-MDBDFL method 10% and 13% compared to FO Lorenz chaotic system and FCMI for Extended Yale Face Database B dataset. Similarly, the precision of the proposed GDH-MDBDFL method is increased by 19% and 22% compared to FO Lorenz chaotic system and FCMI for the Pie dataset.

5. CONCLUSION

A novel method called GDH-MDBDFL is proposed based on FPDE for enhancing the performance of face recognition. To accurately identify the face images, the FPDE algorithm via GDH-MDBDFL is introduced. Primarily, regularized anisotropic diffusion filtering is used to de-noise the input images. Then, Hartigan's clustering based segmentation method is used to segment the images into similar regions based on hamming distance. Also, FPDE is used to identify the diverse types of pertinent features from the images with less time. Lastly, the extracted features are learned through the classification process by using a multidimensional Bregman divergence classifier where the divergence between two images is estimated to recognize the face images. The experiment analysis is performed by using standard benchmark datasets such as the Extended Yale Face Database B dataset and Pie dataset. Both qualitative and quantitative result analysis is presented to verify the effectiveness of the proposed GDH-MDBDFL method. As shown in the simulation results, the proposed GDH-MDBDFL method the higher recognition accuracy, and precision with lesser recognition time compared to the conventional methods.

REFERENCES

[1] Iman S. Badr, Ahmed G. Radwan, El-Sayed M. EL-Rabaie, Lobna A. Said, Ghada M. El Banby, Walid El-Shafai and Fathi E. Abd El-Samie, "Cancellable Face Recognition based on Fractional-Order Lorenz Chaotic System and HAAR Wavelet Fusion", *Digital Signal Processing*, Vol. 116, pp. 1-17, 2021.

[2] Rachid Benouinia, Imad Batiouaa, Khalid Zenkouara,, Azeddine Zahia, Said Najaha and Hassan Qjidaa, "Fractional-Order Orthogonal Chebyshev Moments and Moment Invariants for Image Representation and Pattern Recognition", *Pattern Recognition*, pp. 1-38, 2018.

[3] Sushil Kumar Paul, Saida Bouakaz, Chowdhury Mofizur Rahman and Mohammad Shorif Uddin, "Component-based Face Recognition using Statistical Pattern Matching Analysis", *Pattern Analysis and Applications*, Vol. 89, pp. 1-21, 2020.

[4] Yun Tao Jia, Min Qiang Xu and Ying Zhen Lin, "A Numerical Solution for Variable Order Fractional Functional Differential Equation", *Applied Mathematics Letters*, Vol. 64, pp. 125-130, 2017.

[5] Changpin Li and Fanhai Zeng, "Finite Difference Methods for Fractional Differential Equations", *International Journal of Bifurcation and Chaos*, Vol. 22, No. 4, pp. 1-28, 2012.

[6] N.H. Sweilam, M.M. Khader and H.M. Almarwm, "Numerical Studies for the Variable-Order Nonlinear Fractional Wave Equation", *Fractional Calculus and Applied Analysis*, Vol. 15, pp. 669-683, 2012.

[7] S. Moorthi and S. Karthikeyan, "Deep Gaussian Multivariate Hosmer-Lemeshow Feature Learning via Partial Differential Equation to Face Recognition",

International Journal of Innovative Computing, Information and Control, Vol. 19, No. 1, pp. 1-17, 2023.

[8] Anubha Bhaik, "Detection of Improperly Worn Face Masks using Deep Learning - A Preventive Measure Against the Spread of COVID-19", *International Journal of Interactive Multimedia and Artificial Intelligence*, Vol. 7, No 7, pp. 1-13, 2021.

[9] H. Yi, "Efficient Architecture for Improving Differential Equations based on Normal Equation Method in Deep Learning", *Alexandria Engineering Journal*, Vol.59, No. 4, pp. 2491-2502, 2020.

[10] U.A. Usmani, J. Watada, J. Jaafar, I.A. Aziz and A. Roy, "Particle Swarm Optimization with Deep Learning for Human Action Recognition", *International Journal of Innovative Computing, Information and Control*, Vol.17, No. 6, pp. 1843-1870, 2021.

[11] F.M. Andiani and B. Soewito, "Face Recognition for Work Attendance using Multitask Convolutional Neural Network (MTCNN) and Pre-Trained FaceNet", *ICIC Express Letters*, Vol.15, No.1, pp. 57-65, 2021.

[12] S. Moorthi and S. Karthikeyan, "Deep Gaussian Multivariate Hosmer-Lemeshow Feature Learning via Partial Differential Equation to Face Recognition", *International Journal of Innovative Computing, Information and Control*, Vol.19, No.1, pp. 15-32, 2023.

[13] Soumia Djaghbellou, Zahid Akhtar, Abderraouf Bouziane and Abdeoluahab Attia, "Arabic Handwritten Characters Recognition via Multi-Scale Hog Features and Multi-Layer Deep Rule-Based Classification", *ICTACT Journal on Image and Video Processing*, Vol. 10, No. 4, pp. 2195-2200, 2020.

[14] N.R. Pradeep and J. Ravi, "Machine Learning based Artificial Neural Networks for Fingerprint Recognition", *ICTACT Journal on Image and Video Processing*, Vol. 13, No. 2, pp. 2874-2882, 2022.