Multicultural Education

Volume 12, Issue 10, 2026

An Approach To Enhance Performance Of Age Invariant Face Recognition

Ashutosh Dhamijaa, R. B. Dubeyb

Article Info

Article History

Received:
July 18,2026

Accepted:
October 20,2026

Keywords :
Age Invariance Face Recognition, CNN Architecture, Enhancedactive Shape Model, Handcrafted And Deep Features, Andthe Features Reduction.

DOI:
10.5281/zenodo.23503779

Abstract

In the field of age-invariant face recognition (AIFR), scientific efforts have increased. Matching faces of big age differences is thus a challenge, owing to the significant variation in appearance between young and elderly age. The appearance and form of the face deteriorate with age, making facial recognition the most difficult task. AIFR has become a highly common and difficult chore in recent years. In this discipline, the set of feature extraction and classification algorithms is crucial.In this study, we provide a novel method for combining feature extraction and classification using an upgraded Active Shape Model (ASM) in combination with a specially constructed 7-layered Convolutional Neural Network (CNN). Using the standard dataset LAG, the study approach entails running many tests to evaluate the proposed system's performance. The results show that the suggested method beats state-of-the-art algorithms in face recognition and achieves good accuracy throughout the age spectrum. The exhibited methodology achieves a maximum accuracy of91.76 percent for the LAG database.

Introduction

For many years, face recognition has been a prominent research area in computer vision. By strengthening face recognition models with deep neural networks, deep-learning-based approaches have recently achieved exceptional performance, even surpassing humans in certain instances. To improve face recognition performance, the conventional thinking is to use margin-based measures to boost intra-class compactness and train the models with a large amount of data. For many years, face recognition has been a prominent research area in computer vision. By enhancing the face recognition system, deep-learning-based algorithms have recently achieved excellent performance, even surpassing humans in many cases [1, 2, 3, 4].

In spite of the great progress of generic facial recognition, how to limit the impacts of age variation in existing face recognition algorithms to accurately identify faces in many practical applications such as identifying lost children remains a persistent difficulty. As a result, achieving face recognition without age variation, also known as age invariant face recognition (AIFR), is critical. In the following three areas, however, AIFR remains extremely difficult. First, when the age gap in cross-age face recognition gets high, age variation can have a significant impact on the facial look, compromising face recognition performance. Second, because the facial look substantially changes over time and differs from person to person, face age synthesis is a complex process involving face aging/rejuvenation. Finally, obtaining a big paired face dataset to train a model in portraying faces with natural effects while maintaining identities is impossible [5].

Current AIFR approaches can be classified into two types to address these issues: generative and discriminative models. Given a face image, generative models attempt to combine faces of various ages into a single age group to aid face identification. GANs [6] have recently been employed to improve the image quality of synthesized faces; they typically use one-hot encoding to designate the target age group. The one-hot encoding, on the other hand, represents age group-level face transformations while neglecting identity-level individualized patterns, resulting in unanticipated distortions [7, 6, 8, 9, 10, 11, 12, 13, 14].

As a result of the unappealing synthesized faces and unanticipated shifts in identity, AIFR's performance cannot be greatly enhanced. Discriminative models, on the other hand, concentrate on extracting age-invariant characteristics by disentangling identity-related information from mixed information, so that only identity-related information is expected for face recognition systems. Despite their promising performance in AIFR, they are unable to give users, such as police officers, with visual results as generative approaches for further verifying identities, which can undermine model interpretability in many practical applications [15, 16].

The reduction of dimensionality is another big problem with the introduction of deep learning methods without losing critical data is the basic information required for classification [17]. Few techniques aim to empirically infer variable predictability [18] and reduce managed feature selection, but that often leads to a lack of clarification in detail. The theory is focused on human actions, in which machine-learning algorithms take

action to minimize potential costs. Several researchers have concentrated on designing parallel, adaptive, and deep learning frameworks in recent years [19,20]. The focus has recently been updated to improve the way we think about GPUs. However, GPUs are notorious for leakage currents that abstract the conceptual execution of profound research models on mobile devices [21]. Field-Programmable Gate Arrays (FPGAs) may be used as deep learning accelerators to allow the configuration of data-access pipelines to deliver significantly better results [22].

Despite recent advances in deep neural network (DNN) approaches, face recognition is still a very challenging problem due to varying shapes and appearances. An Active Shape Model (ASM) using prior knowledge of the object about shape and appearance increases the robustness of the results. To take advantage of these benefits, this paper proposes an enhanced ASM framework [5].

The parametric deformable models like the ASM [23] can represent large shape and appearance variations of the object of interest and use. The ASM describes the contour of the object of interest by a set of model points. This finds specific positions on the object of interest such as corner of the lips on a face called as landmarks, through an iterative process describe using two main criteria: i) searching the image for those landmarks, from which candidate positions for the location, and ii) estimating the parameters that best fit the model to those observations [24]. A major limitation of ASM is searching for the landmarks which hamper the estimation of the model parameters and degrading the accuracy of the fitting process.

This paper addresses the above issues by: 1) proposing an enhanced ASM approach to extract features from 2D search regions using handcrafted and deep image features 2) proposing 7-layer CNN architecture with the use of a smaller image size of 32x32 pixels to minimize delay time and space complexity. The proposed framework is tested on standard dataset LAG. The findings showed that the suggested method outperforms existing state-of-the-art recognition methods.

The remainder of the research paper is as follows: Section-II highlights the related work done in this area. Section-III demonstrates the CNN framework. Section-IV points out the implementation of this approach. Section-V carries out the study findings and interpretation. The conclusions are eventually made in Section VI.

1. Related Work

Face recognition that isn't affected by age (AIFR). Previously, age variation was mainly addressed by disentangling age-invariant characteristics from mixed data. For example, to factories mixed characteristics and reduce age variance in identity-related features, used hidden factor analysis (HFA). [25] used the latent factor guided convolutional neural network to extend HFA into a deep learning framework (LF-CNN). Simultaneously, added an age estimation duty to help guide the AIFR. Recently, CNN-based discriminative approaches for AIFR have shown promising results. By partitioning the facial embedding's into two orthogonal components, OE-CNN implemented a modified softmax loss for AIFR, so that the identity- and age-related characteristics are represented as the angular and radial directions, respectively. Similarly, on the premise that the two components are uncorrelated, DAL performed feature decomposition in an adversarial manner [26, 27, 28, 16, 25, 29].

A paper [30] uses a cGANs-based model with cross-age domain adversarial training to extract age-invariant representations to accomplish both tasks at the same time. It does, however, cause modest variations over smoothed faces.Synthesis of facial age (FAS); physical model-, prototype-, and deep generative model-based methods are the three types of FAS methods now available. Physical model-based approaches mechanically model changes in appearance over time, but they are computationally expensive and require a large number of paired photos of the same individual taken over a lengthy period of time. Face aging/rejuvenation is achieved using the average of faces in each age group in prototype-based approaches, hence identity is not adequately preserved. For this job, deep generative model-based approaches use a deep neural network. Recurrent face ageing (RFA) [31], for example, used a recurrent neural network to simulate the intermediate transition phases of age progression/regression, relying on which a smooth face ageing process could be obtained. Many recent studies use generative adversarial networks (GANs) [6], particularly conditional GANs (cGANs) [32], to improve the visual quality of synthesized faces and train models with unpaired age data, inspired by their great capabilities in creating high-quality photos. For example, by travelling on a low-dimensional face manifold, [14] used a conditional adversarial autoencoder (CAAE) to achieve both age progression and regression. Perceptual loss was established by to maintain identities during face aging/rejuvenation. To highlight the ageing details, [13] built a discriminator with a pyramid construction [6, 33, 34, 32, 35, 36, 37, 38, 39, 31, 12, 13, 14].

2. Overview of Convolutional Neural Network (CNN) approach

CNN is a popular machine-learning method in the area of profound learning. CNN are specialized in large sets of different objects. CNN can learn rich depictions for a wide range of images from these vast collections. Deep-learning architectures consist of three main layers, for example, the input layer, hidden layers, and the output layer. The number of concealed layers determines the architectural scope. It involves learning various non-linear

functions according to the form of hidden layers. There are different hidden layers in a basic deep learning model.
The input layer determines the form and dimension of data that can be handled by CNN. The following step is the description of the network's middle layers. The core layers consist of regular, rectified linear blocks of ReLU and pooling layers. These three layers form the core blocks of convolution neural networks that establish filter weights and update network training. The ReLU layer introduces non-linearity to the network, allowing network non-linear functions to be approximated. As a consequence, the image pixels are mapped to the semanticized content.The pooling layers down sample data as the network runs. Pooling layers can be used sparingly in a network of multiple layers to prevent the down sampling of the data early in the network. By repeating these three simple layers, a deeper network can be built. To prevent premature sampling of the information, the number of pooling layers should be decreased. Usually, the final layers of a CNN consist of fully connected and softmax layers. The following describes the 7 levels of CNN architecture [40-42]:

***3.2.1 Input image layer*:** A layer of image information enters images into the network and normalizes the data.

3.2.2 Convolutional layer: The data is filled with a 2-D convolution layer. It bends the input vertically and horizontally pushing the filtering over the input. It measures the weight and the input point product and applies a choice. The machine measures the measurements of the zero-padding during the testing so that the layer output is the same as the data.

***3.2.3 Rectified Linear Unit (ReLU) layer*:** A threshold function is used by a ReLU layer for every input element that is set to zero by a value less than zero. It is known as linear for values greater than null, which means that when training a neural network with back propagation, a linear activation function has many desirable features. However, this function is non-linear since negative values are always represented as zero.

***3.2.4 Max pooling layer*:** A maximum pooling layer divides the input intorectangular areas of pooling and determines the limit of each field. Another building block of a CNN is a pooling layer. The key function of the max-pooling layer is to steadily reduce the image space. It allows us to reduce the number of parameters and the number of network computations. Every function map has a different pooling layer. Max pooling is the most common solution to pooling.

***3.2.5 Fully connected layer*:** The input is multiplied by a fully connected layer by weight and a weight vector is then added. The aim of a fully connected layer is to take and organize the effects of the convolution process into a mark which requires the use of a fully linked layer for classification purposes at the end of the network. In relation to pooling and convolution, it is called a global process. It gathers feature-extraction data and analyses the performance of all previous layers globally. This results in a non-linear mix of chosen features used for data classification.

***3.2.6 Softmax layer*:** The input is protected by a softmax layer. In general, softmax is known to be an active element, like sigmoid or ReLU. The activation function functions as a judgment and helps to understand dynamic patterns. Choosing a fitting triggering mechanism speeds up the learning process. No loss function, however, softmax is used to render the performance of a neural net consistent with cross-entropy or negative log-like loss functions.

***3.2.7 Classification Layer*:** Build an output layer classification.

The architecture of CNN is designed as follows:

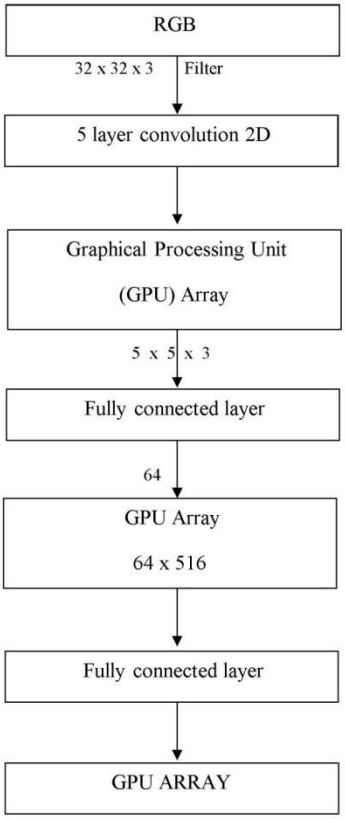


Fig.1: Block diagram of CNN

In CNN architecture, the multiple layer connection was utilized consisting of various hidden layers. The fully connected layer is the hidden layer as shown below:

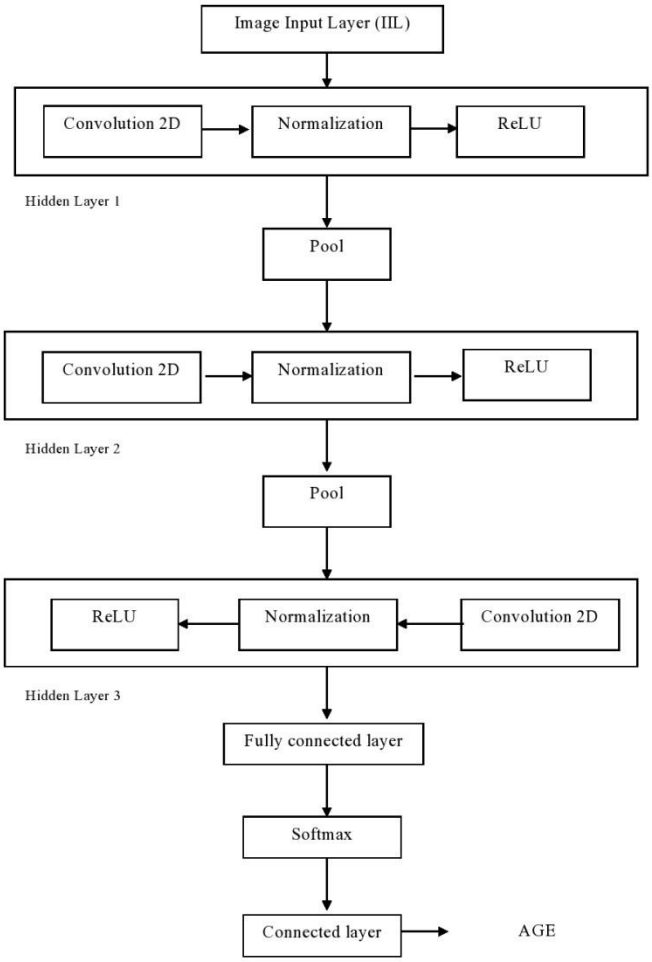


Fig. 2: Hidden layer architecture

3. Implementation of proposed methodology

This section explains how to address AIFR problems using an upgraded ASM paired with a specially constructed CNN. A schematic block diagram of the approach is shown in Fig. 3. The three essential components of the technique are preprocessing, feature extraction, and classifications. A number of simple pre-processing processes are employed to improve device efficiency while focusing on data pre-processing. Feature extraction captures the required function descriptors using enhanced ASM and CNN rather than manually gathering them. For age classification, we used the LAG dataset. The image size of each dataset has been modified to 32x32x3. Extra steps are taken to establish whether the image is RGB or grayscale. There are two portions to the dataset: cross-validation training and testing, as well as a traditional train (70%) to test (30%) ratio. For feature extraction and faster performance, the qualified MUCT dataset is employed.

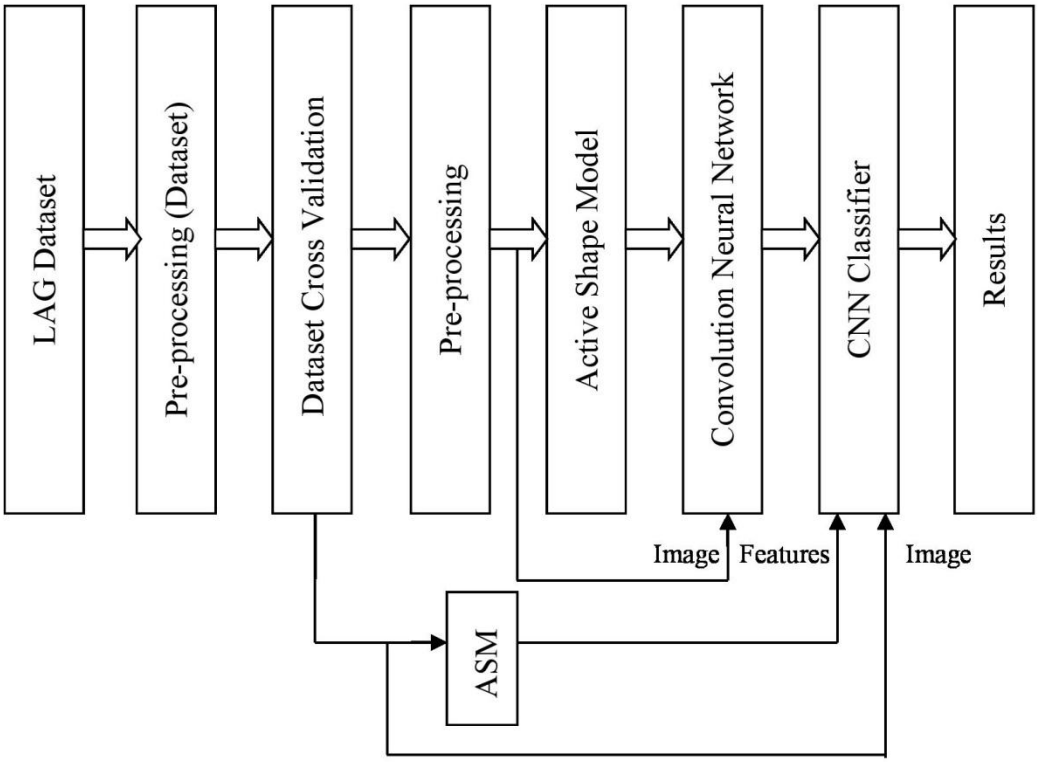


Fig. 3: Block diagram ofproposed methodology

4.1. Image preprocessing

Traditional databases include images of various sizes, which might help with issue categorization. Image preprocessing aids in the normalization of the dataset. The RGB image is then converted to grayscale and shrunk down to 32x32x3 pixels. Fig. 4 shows example of preprocessed images from the LAG dataset.

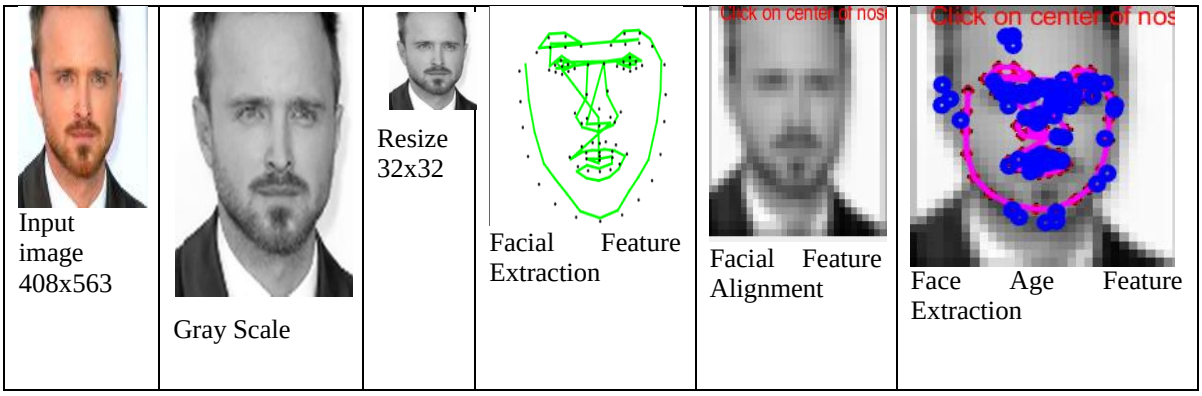


Fig. 4: Stepwise preprocessing of sample image from LAG dataset

4.2. Enhanced Active Shape Model (ASM)

The contour of the object of interest by a set of N model points is given by $x = \left[x_1^T, \ldots . x_N^T \right]^T$, where $x \in \mathbb{R}^{2N\times 1}$, and $x_i \in \mathbb{R}^{2\times 1}$ are the coordinates of the i-th point. In the ASM structure [23], a contour can be represented in the shape space by the average shape $\bar{x} = \left[\bar{x}_1^T, \ldots . \bar{x}_N^T \right]^T \in \mathbb{R}^{2N\times 1}$, deformed by a linear combination of the K main modes of deformation $\mathbf{D} \in \mathbb{R}^{2N\times K}$,

$$\tilde{x}_i = \bar{x}_i + D_i\, b \ , \qquad (1)$$

where $\mathbf{b} \in \mathbb{R}^{k\times 1}$ contains the deformation coefficients that weight each deformation mode and $D_i \in \mathbb{R}^{2\times K}$ contains the lines of D associatedwith each model point $\tilde{x}_i$.

The pose of the object is then represented by a global pose transformation, with parameters **a** and**t**. Here, $t = \left(t_1, t_2\right)^T$ computes the translation of the contour, and a = (s cos (ϕ), s sin (ϕ))^{T}computes its scaling s, and its rotation ϕ. Similarly, each point in the shape space, $\tilde{x}_i$, can then be mapped into the image space x_i, by

$$x_i = A\tilde{x}_i + t$$

$$= \begin{bmatrix} s & \cos(\phi) & -s & \sin(\phi) \\ s & \sin(\phi) & s & \cos(\phi) \end{bmatrix} \begin{bmatrix} x_{i2}^1 \\ x_i \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \end{bmatrix} \qquad (2)$$

From Eqns. (1) and (2), the position of each shape model point is given by

$$x_i = A\left(\tilde{x}_i + D_i b\right) + t\,, \qquad (3)$$

Any shape in the image space is represented by this. As a result, there are two sorts of parameters in an object's shape model: 1) global pose transformation parameters (**a**, **t**) and 2) local deformation parameter b.ASM employs prior information gained from a group of annotated image called the training set to recognize the shape of an item. The alignment of the training set and the computation of its shape statistics are the two stages of this learning technique [5, 43].

4.3. Image features

The suggested methodology employs an image feature extraction procedure that differs from a normal ASM framework in two ways: 1) We use 2-D search regions instead of a 1-D search line, and 2) we use alternative image characteristics to extract measurements within the 2-D search regions instead of existing edge detection. Deep features are extracted using CNN in addition to handcrafted Histogram of Oriented Gradients (HOG) [44], Scale-Invariant Feature Transform (SIFT) [45], and constant and variable texture templates Fixed Texture Templates (FTT) and Variable Texture Templates (VTT).

4.4 Optimization of the model

When variations are present, it is difficult to evaluate model parameters. It's impossible to know which measures are correct and which are not. It's frequently misguided, and it doesn't work well in a model. To reduce the variations in the parameters of the ASM, we adopt an efficient model based on the generalized expectation maximization algorithm [5, 46].

In light of deviations, the model parameters can only be changed to a certain extent. Being the model doesn't tell which deviations are real and which aren't. It's out of date, yet it can be useful to some extent. We employ a sophisticated model based on Generalized Expectation Maximization for effective deviation handling during ASM parameter optimization [5, 46].

4.5. Network architecture

CNNs are designed to process a 2-D image. A CNN has three major layer types: (a) convolution layers, (b) sub-sampling layers, and (c) output layers. Network layers are structured in the feed-forward structure: a sub-sampling layer accompanies every convolution layer and a fullyconvolution layer is followed by an output layer. The convolution and sub-sample layers are known as 2-D, while the output layer is treated as 1-D. Each 2-D layer has a several planes on CNN. A plane is composed of neurons arranged in a 2-D array. The output of a plane is a feature map. We hereby proposed a 7-layer architecture with 3 layers of convolution (C1, C3, C5),

2 layers of pooling (P2, P4), and 2 fully linked layers of output (F6, F7) [40, 47]. The architecture for the proposed methodology is shown in Fig. 5.

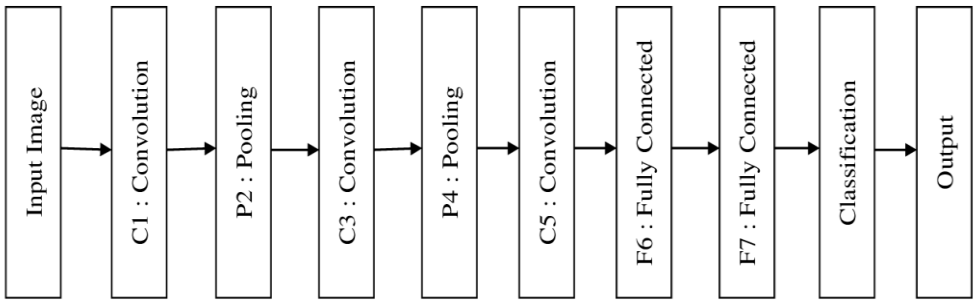


Fig. 5: 7-Layer CNN Architecture

A 5x5 filter is convolved with a 32x32 pixel image to create a 28x28 pixel filtered image. It is divided into six separate planes. These 6 planes produce 6 different feature maps, resulting in a performance of 28 x 28 x 6 for the first convolution layer C1. The second layer of the sub-sampling layer is Layer 2. The regions are 2x2 pixels in size, and the pooling process is summarized. It results in a two-fold reduction in both dimensions and a 14x14x6 matrix in a typical map. C3, the third convolution layer, is made up of a filter with the same 5x5 scale as the other two. This tier contains 16 independent 10x10 planes. Layer 4 reflects the layer 4 sub-sampling with a 5x5x16 matrix and a role map decreased by a factor of 2 by block 2. Finally, the convolution layer C5 provides a 120-scalar-value feature diagram. Lastly, the next layer has two fully linked layers (F6 and F7), each of which is linked to all inputs. F6 consists of 84 neurons, while F7 consists of ten. The output of the last completely linked layer F7 produces the classification output.

4.5.1 *Convolution layer*

Each plane of the convolution layer has one or earlier layer maps. Convolution mask is used as a 2-D weight matrix for the related relation of adjustable inputs. In each plane, the convolution is determined between its 2-D inputs and its mask. The output of the convolution layers with an adjustable scalar, known as bias, is applied. Finally, an activation function is added to get the output of the plane. This performance is regarded as a feature map for each plane. One or more maps can be present in a convolution layer. Each of these maps is related to precisely one step of the next layer pooling layer. Each plane in the last convolution layer is connected to an accurate characteristic map of the previous layer. Each plane generates one scalar output in the convolution layer. All outputs from all planes are provided to the output layer. This layer is meant to strip low-level features like edges and texture. The convolution layer feature map is determined as:

$$y_n^l = f_l\left(\sum_{m\in V_n^l} y_m^{l-1} \otimes w_{m,n}^l + b_n^l\right) \qquad (4)$$

where $w_{m,n}^l$ is the convolution mask, b_n^l is the bias term, and V_n^l is the list of planes [40].

4.5.2 Pooling (sub-sampling) layer

The dimensionality of each feature map is minimized by the spatial pooling to preserve the most important details. Spatial pooling will take three types: Max pooling-takes the largest element, average pooling-takes the elements by average, and sum pooling-takes all the elements in amount. The key role of pooling is to increasingly reduce the spatial scale of the input monitor. It helps to render smaller and more convenient data representations. A pooling and prior convolution layers provide the same amount of planes. The findings obtained are then passed to generate the outputs. This characteristic map is bound to one or more planes on the next convolution layer. It enhances the production of the convolution layer to local distortions. The pooling layer feature map is determined as

$$y_n^l = f_l\left(Z_n^{l-1} \times w_n^l + b_n^l\right) (5)$$

where Z_n^{l-1} is a matrix obtained by summing all four pixels of a block, w_n^l is the weight and b_n^l is the bias term [48].

4.5.3 Activation function

The activation function is a decision-making function that offers awareness of dynamic patterns. The selection of a suitable activation mechanism will speed up the learning process. Equation determines the activation mechanism for a convolved feature map (6)

$$T_l^k = g_a\left(F_l^k\right) (6)$$

In the above equation, F_l^k is an output of a convolution, which is assigned to activation function $g_a(.)$ that adds non-linearity and returns a transformed output T_l^k for l^{th} layer. Here, ReLU and its variants are preferred as they help in overcoming the vanishing gradient problem [49-50].

4.5.4 Batch normalization

Batch normalization is used to resolve the internal covariance change problems in feature maps. The shift in internal covariance is a change in the distribution of hidden unit values which slows down convergence and needs to be carefully initialized. Batch normalization for a transformed feature-map Eqn. (7)

$$N_l^k = \frac{F_l^k - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}} \quad (7)$$

In equation (7), N_l^k represents normalized feature-map, F_l^k is the input feature map, μ_B, and σ^2_B depict mean and variance of a feature–map for a mini-batch respectively. To avoid division by zero □ □is added for numerical stability. Batch normalization unifies the distribution of feature map values by setting zero mean and unit variance. Batch normalization often plays an essential part in smoothing gradient flow and serves as a regulatory factor. It allows us to boost network generalization [51].

4.5.5 Output layer (fully connected layer)

The output layer is created with a sigmoid neuron in CNN. The outputs of this layer are normally the network outputs. In the output layer, a conventional multi-layer vision uses the softmax activation function. The fully linked layers catch the associations between the characteristics of separate sections of the face such as the structure and position of the eyes and mouth. The convolution and pooling layers in conjunction are used for the extraction of features and classification with fully connected layers. The neuron's performance is measured as:

$$y_n^L = F^L\left(\sum_{m=1}^{N^{L-1}} y_m^{L-1} w_{m,n}^L + b_n^L\right) (8)$$

where N^L is the number of outputsigmoidalneurons, $w^L_{m,n}$ is weight from feature map m of the last convolution layer to neuron n of the output layer, and b^L_n is the bias of neuron n associated with layer L [40]. Finally, the output of the CNN classifier is obtained that helps in the identification of an image.

4. Experimental results and analysis

The AIFR-CNN experimental results are explained in this section. All tests are carried out with MATLAB 2017a (64-bit) with Intel i5 6600U 8 GB RAM 2 GB NVidia GPU. Extensive tests on LAG dataset are carried out to assess the approach's effectiveness. There are significant variances in posture, speech, and lighting conditions among the dataset.

5.1 Datasets

5.1.1MUCT (Milborrow University of Cape Town) database

The MUCT database has 3,755 faces with 76 manual points of reference. The MUCT database has a larger variety in illumination, age, and race than current publicly accessible 2D manually-marked datasets. As an illustration of the database method, we see that the Active Shape Model trained on MUCT data is a little more stable than the one trained on XM2VTS data [52].

5.1.2LAG (Large Age-Gap) dataset

The large age-gap data set (LAG) was introduced recently to research the issue of cross-aging face recognition. It contained all the images taken in the wild, with an era of 0-80 years. The data collection is generated through

a search by Google. It includes 3,828 images with 1,010 personalities. For each identity, at least one child and one adult image are available [48]. Fig. 6shows certain LAG database sample images with age values.



Fig.6: Test LAG dataset images with age values

5.2 Experiments on a LAGdataset

The LAG dataset is recently released to investigate the challenge of cross-age face recognition. Manually checking and removing noisy and duplicate images is the next stage. The ASM-CNN model is then trained on the MUCT data set to improve performance. To fully examine our suggested model, we study the performance of our algorithm in relation to the local bases of the codebook, as specified in Section 4. We use a two-fold cross validation approach to assess performance. The average recognition accuracy is calculated after subjects are alternately assigned to the first and second folds. The training set is made up of images that have been flipped horizontally and vertically. The horizontally rotated vertically flipped images are further rotated.

We used a total of 1118 images from the LAG dataset for our observations, 783 images are used for training, and 335 images for testing. Following that, after all of the pre-processing processes, we employed images with a scale of 32x32. There are a lot of head-posing combinations in these images. The result parameter is calculated using Rank-1 identification. According to the results, 307 out of 335 images are correctly identified. It represents Rank-1 recognition of 91.76%. Next, we employed the same procedure and quantity of images as before, but with a 64x64 images scale. In this experiment, 302 out of 335 images are correctly identified. It has a Rank-1 recognition accuracy of 90.15%. Due to the constrained existence of the network design, the image size 64x64 was chosen but it is not adequately powerful for this image scale. In the experiment, we utilized a 32x32 pixel image because it generates better results.

The efficiency of our proposed model is compared to the state-of-the-art high-dimensional LBP [53] and several similarity metric learning methods, including Cosine similarity [54], the one-shot similarity kernel [55], the joint Bayesian [56], and sub-SML [57]. All of these methods of similarity metric learning are employed in the features taken from the DCNN [58] (trained in the data set of the CASIA-Web Face [59]). Table 1 summarizes the findings.

Table 1: Comparative findings for the LAG database average Rank-1 recognition rate.

S. No.	Algorithms	Rank -1 Recognition rates
1	DCNN [60] + Joint Bayesian [55]	66.33%
2	DCNN [58] + OSS [57]	66.42%
3	HDLBP [53]	71.53%
4	DCNN [61] + CARC [62]	74.82%
5	DSIFT+LBPD + NN [63]	79.88%
6	DSIFT+LBPD + LR [63]	80.00%
7	Bianco et al. [48]	84.95%
8	Deep features + LR [63]	89.44%
9	Deep Features + LR (Hampel) [63]	90.50%

10	Deep features + NN [63]	91.00%
11	Deep Features + NN (Hampel) [63]	91.63%
12	**Demonstrated ASM-CNN**	**91.76 %**

For the LAG dataset, ASM-CNN achieves a maximum recognition rate of 91.76%. Furthermore, the effectiveness of our method is evaluated using LAG dataset, as shown in Table 1, Table 2, and Fig.7. Some examples of accurately detected matches among those on which our solution has a high level of confidence are shown in Fig. 8. Fig.10 shows some examples of successfully recognized matches on which our solution isn't so sure. It's clear from the samples that matching pairings on which our solution is quite certain have the same pose and expression, implying that more powerful pose-normalization algorithms could enhance verification accuracy even more.

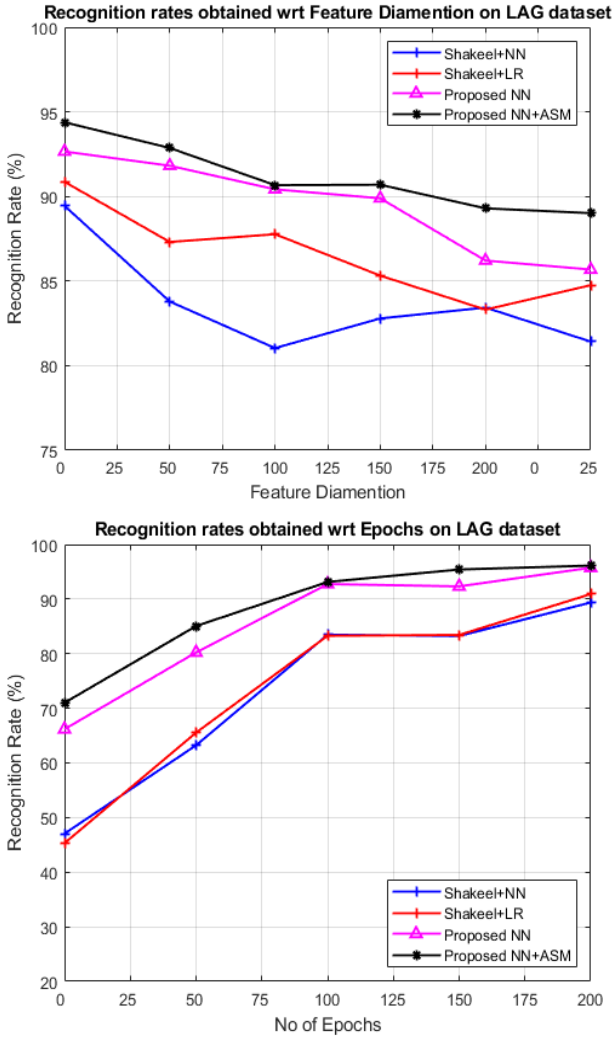


Fig.7: Recognition rates obtained by LAG data set. (a) Feature of dimensions and (b) Number of epochs



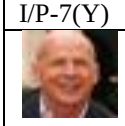
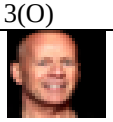
I/P-7(Y)	1(O)	3(O)	6(O)	10(Y)	14(Y)
					
I/P-6(O)	2(O)	5(O)	8(Y)	10(Y)	15(Y)

Fig. 8: The test correctly recognized findings utilizing LAG dataset. The numbers below display the age of the person. Y stands for the young and O stands for the old

Table 2: Comparative Rank-1 recognition of various LAG dataset image sizes (All images with head pose variation)

Image Size	Training Images	Testing Images	Rank-1 Recognition
32x32	783	335	91.76%
64x64	783	335	90.15%

5.5 Overall benchmark comparisons

Here, we equate our theoretical approach to the present state-of-the-art. Various factors such as head location changes, facial features, and aging symptoms render it impossible to identify the face. Many researchers have suggested their methodologies to solve the issue of facial recognition by aging. It normally involved measures such as facial recognition, preprocessing, features extraction, and classification. The technique efficiency depends directly on the algorithms used to retrieve and identify the feature. Although CNN is a powerful tool, the number of layers, the number of neurons, and the scale of the image supplied to the CNN architecture are difficult to ascertain. Sadly, no route or solution is available. However, no one concentrated on these problems, instead of creating their own design. In our design, we utilize the same method to determine the number of layers, their dimensions, and the size of the image provided to CNN.Our experimental findings revealed that our suggested approach is stable in the presence of noise variations. For the LAG data set, there is just a 4-5 percent reduction in recognition output.

Table 3: Comparative analysis of various datasets.

S. No.	Dataset	Accuracy (%)
1	LAG	89.65 %
2	LAG (trained with MUCT)	91.76 %

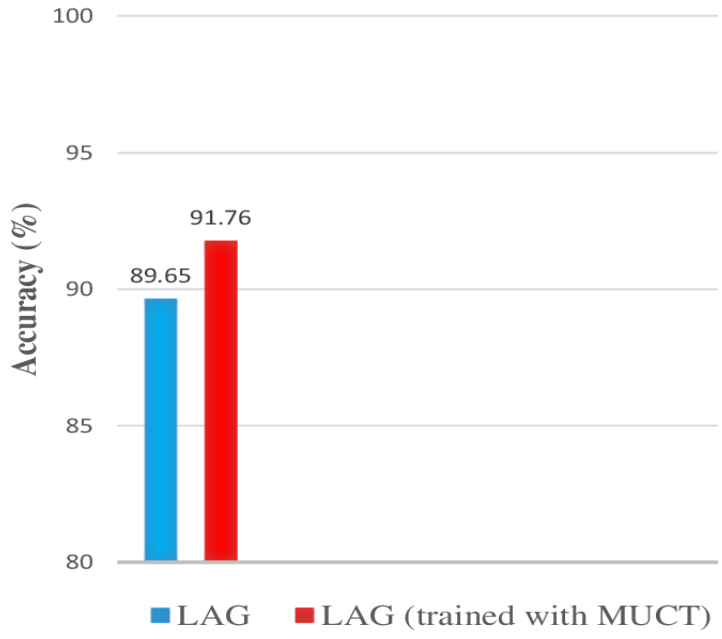


Fig. 9: A comprehensive-comparative analysis for effect of ASM-CNN on accuracy of LAG dataset.

Table 3 shows that the accuracy of LAG and LAG (trained with MUCT) 89.65% and 91.76 % respectively. Fig. 9 shows a comprehensive-comparative investigation of the influence of ASM-CNN on the accuracy ofLAG dataset. On LAG dataset, experimental results reveal that the proposed method outperforms state-of-the-art approaches and achieves excellent accuracy in face recognition across age.

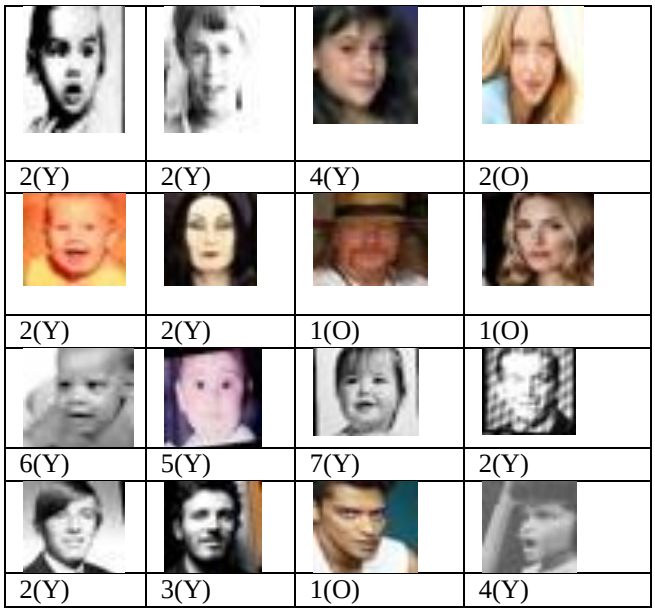


Fig.10: Some wrongly accepted findings using the LAG dataset

Fig. 10 revealed some of the missed LAG Rank-1 recovery performance. The findings suggested that intra-class disparities and inter-class correlations occurred in the dataset. Manually it was often challenging to classify the groups since some appear identical to others. Fig.10 revealed that certain improperly accepted findings have been reported using the LAG dataset. The numbers below indicated the person's age. The suggested approach is used to report cumulative LAG images in datasets 1118 and 16 images with improperly recognized effects.In this analysis, we used images of 32x32 formats after preprocessing. It requires less time to measure and less room for a huge database. This increases the efficiency of dataset as seen in Table 4.

It is established from the experiments that:

a. In Rank-1identification, ASM-CNN is superior accessible approaches with no challenging preprocessing measures such as histogram standardization and head adjustment.
b. It has a stronger performance on image scale 32x32 than the image size 64x64. When the image scale grows, it takes more time to implement.
c. It provides stronger performance as a classifier with ASM-CNN over CNN.

5.5.1 Computational time analysis

Our system comprises three key stages, preprocessing, features extraction, and classification. In this portion, the computing time of a single image and complete dataset are seen in seconds in Table 4.

Table 4: The computational time analysis

Datasets	**Single image**	**Whole data set**
LAG	1.34 sec	317.540 sec

5.5.2 Comparing with state-of-the-art algorithms

Comparison with other age-invariant face recognition algorithms is made to check the consistency of the proposed method for age-invariant face recognition. The findings are summarized in Tables 1, 2, 3 and 4. It is seen that certain approaches cannot be evaluated as various datasets or different sample amounts havebeen used in each study or on various laboratory settings. As can be seen in Table 1, Table 2, Table 3 and Table 4, the ASM-CNN suggested greatly enhanced the accuracy of recognition over other comparable schemes.

5. Conclusions

We introduced in this paper a new approach ASM-CNN framework for the age invariance face recognition. The standard LAG image dataset hasbeenexperimented. In this method, our goal was to build a transparent network by using fewer layers of smaller image sizes (32x32). This method has retained simplicity, as feature extraction, no separate algorithm is required. The findings revealed that the identification accuracies on these dataset arebetter than the existing state of the art. The resized 32x32-pixel image resolution showed greater performance than those of 64x64-pixels.The combination of CNN with ASM demonstrated a substantial increase in classification accuracy.

More features will be explored in the future to make the method more reliable, especially where a wide age gap is taken into account. We will also concentrate on our future work on creating a new convolutional layer that can be built into the feature recovery network to build all feature mergers.

Conflict of Interest: The authors declare that they have no conflict of interest.
Funding: NIL (No funding Grant)

References

K. He, X. Zhang, S. Ren and J. Sun, Deep residual learning for image recognition, In IEEE Conf. Comput. Vis. Pattern Recog. (2016) 770–778.
A. Krizhevsky, I. Sutskever and G. E. Hinton, Imagenet classification with deep convolutional neural networks Communications of the ACM 60(6) (2017) 84-90.
K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition, In Proc. Int. Conf. Learn. Represent (2015)1–14.
Y. Wen, K. Zhang, Z. Li and Y. Qiao, A discriminative feature learning approach for deep face recognition, In Eur. Conf. Comput. Vis. (2016) 499–515.
D. O. Medley, C. Santiago and J. C. Nascimento, Deep active shape model for robust object fitting, IEEE Transactions on Image Processing (2019) 1-14.doi:10.1109/tip.2019.2948728
I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, Generative adversarial nets, In Proc. Adv. Neural Inf. Process. Syst. (2014) 2672–2680.
X. Geng, Z. H. Zhou and K. S. Miles, Automatic age estimation based on facial aging patterns, IEEE Trans. Pattern Anal. Mach. Intell. 29(12) (2007) 2234–2240.
A. Lanitis, C. J. Taylor and T. F Cootes, Toward automatic simulation of aging effects on face images, IEEE Trans. Pattern Anal. Mach. Intell. 24(4) (2002) 442–455.
Q. Li, Y. Liu and Z. Sun, Age progression and regression with spatial attention modules, In AAAI, 2020.
Y. Liu, Q. Li and Z. Sun, Attribute-aware face aging with wavelet-based generative adversarial networks, In IEEE Conf. Comput. Vis. Pattern Recog. (2019) 11877–11886.
U. Park, Y. Tong and A. K Jain, Age-invariant face recognition, IEEE Trans. Pattern Anal. Mach. Intell., 32(5) (2010) 947–954.
Z. Wang, X. Tang, W. Luo and S. Gao, Face aging with identity-preserved conditional generative adversarial networks, In IEEE Conf. Comput. Vis. Pattern Recog. (2018) 7939–7947.
H. Yang, D. Huang, Y. Wang and A. K Jain, Learning face age progression: A pyramid architecture of gans, In IEEE Conf. Comput. Vis. Pattern Recog. (2018) 31–39.
Z. Zhang, Y. Song and H. Qi, Age progression/regression by conditional adversarial autoencoder, In IEEE Conf. Comput. Vis. Pattern Recog. (2017) 810–5818.
D. Deb, D. Aggarwal and A. K Jain, Finding missing children: Aging deep face features. arXiv preprint arXiv:1911.07538, 2019.
Y. Wang, D. Gong, Z. Zhou, X. Ji, H. Wang, Z. Li, W. Liu and T. Zhang, Orthogonal deep features decomposition for age-invariant face recognition, In Eur. Conf. Comput. Vis. (2018) 738–753.
E. W. Newell and Y. Cheng, Mass cytometry: Blessed with the curse of dimensionality, nature immunology 17(2016) 890–895.
L. Breiman, Statistical modeling: The two cultures, quality control and applied statistics 48(1) (2003) 81–82.
G. E. Dahl, D. Yu, L. Deng and A. Acero, Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition, IEEE Trans. on Audio, Speech and Lang. Process.20(1) (2012) 30–42.
G. Hinton, L. Deng, D. Yu, G. Dahl, A.R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath and B. Kingsbury, Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, IEEE Sig. Process. Magaz.29(6) (2012) 82–97.
S. Hong and H. Kim, An integrated GPU power and performance model, In: International Symp. on Comp. Arch., ACM 2010, pp. 280–289.

Q. Yu, C.Wang, X. Ma, X. Li and X. Zhou, A deep learning prediction process accelerator based FPGA, In: 2015 IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (2015)1159–1162.
T. F. Cootes, C. J. Taylor, D. H. Cooper and J. Graham, Active shape models - their training and application, Computer vision and image understanding 61(1) (1995) 38–59.
J. A. Nahed, M.P. Jolly and G.Z. Yang, Robust Active Shape Models: A Robust, Generic and simple automatic segmentation tool, in: Larsen R., Nielsen M., Sporring J. (eds) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2006. MICCAI 2006. Lecture Notes in Computer Science, vol. 4191. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11866763_1
Y. Wen, Z. Li, and Y. Qiao, Latent factor guided convolutional neural networks for age-invariant face recognition, In IEEE Conf. Comput. Vis. Pattern Recog. (2016) 4893–4901.
D. Gong, Z. Li, D. Lin, J. Liu, and X. Tang, Hidden factor analysis for age invariant face recognition, In Int. Conf. Comput. Vis. (2013) 2872–2879.
W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj and L. Song, Sphereface: Deep hypersphereembedding for face recognition, In IEEE Conf. Comput. Vis. Pattern Recog. (2017) 212–220.
H. Wang, D. Gong, Z. Li and W. Liu, Decorrelated adversarial learning for age-invariant face recognition, In: IEEE Conf. Comput. Vis. Pattern Recog. (2019) 3527–3536.
T. Zheng, W. Deng and J. Hu, Age estimation guided convolutional neural network for age-invariant face recognition, In IEEE Conf. Comput. Vis. Pattern Recog. Worksh. (2017) 1–9.
J. Zhao, Y. Cheng, Y. Cheng, Y. Yang, F. Zhao, J. Li, H. Liu, S. Yan and J. Feng, Look across elapse: Disentangled representation learning and photorealistic cross-age face synthesis for age-invariant face recognition. In: AAAI 33 (2019) 9251–9258.
W. Wang, Z. Cui, Y. Yan, J. Feng, S. Yan, X. Shu, and N. Sebe, Recurrent face aging, In: IEEE Conf. Comput. Vis. Pattern Recog. (2016) 2378–2386.
M. Mirza and S. Osindero, Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784, 2014.
H. Huang, S. Chen, J. Zhang and H. Shan. PFA-GAN:Progressive face aging with generative adversarial network. TIFS, 2020.
I. Kemelmacher-Shlizerman, S. Suwajanakorn and S. M Seitz, Illumination-aware age progression, In: IEEE Conf. Comput. Vis. Pattern Recog. (2014) 3334–3341.
C. N. Duong, K. G. Quach, K. Luu, N. Le and M. Savvides, Temporal non-volume preserving approach to facial age-progression and age-invariant face recognition, In: Int. Conf. Comput. Vis. (2017) 3735–3743.
N. Ramanathan and R. Chellappa, Modeling age progression in young faces. In: IEEE Conf. Comput. Vis. Pattern Recog. 1(2006), 387–394.
N. Ramanathan and R. Chellappa, Modeling shape and textural variations in aging faces, In IEEE Int. Conf. on Auto. Face & Gesture Recog. (2008)1–8.
D. A. Rowland and D. I. Perrett, Manipulating facial appearance through shape and color, IEEE Comp. Graph. and Applications 15(5) (1995)70–76.
J. Suo, X. Chen, S. Shan, W. Gao, and Q. Dai, A concatenational graph evolution aging model, IEEE Trans. Pattern Anal. Mach. Intell. 34(11) (2012)2083–2096.
S. L. Phung, A. Bouzerdoum, MATLAB library for convolutional neural networks, Technical Report, Visual and Audio Signal Processing Lab University of Wollongong, 2009
N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, Dropout-A simple way to prevent neural networks from Overfitting, Journal of Mach. Learn.Research15(2014) 1929-1958.
M. Lin, Q. Chen and S. Yan, Network in network, Neural and Evolutionary Computing (2013) 1–10.
D. Medley, C. Santiago and J. C. Nascimento, Robust feature descriptors for object segmentation using active shape models, In: Proceedings of the Int. Conf. on Adv. Concepts for Intell. Visi. Sys. (2018)163–174.
N. Dalal, W. Triggs, Histograms of oriented gradients for human detection, In: Proceedings of the IEEE Conference on Comp.Visi.and PatternReco. (2005) 886–893.
D. G. Lowe, Distinctive image features from scale-invariant key points, Int. Journal of Comput. Visi. 60(2) (2004) 91–110.
R. M. Neal, G. E. Hinton, A view of the EM algorithm that justifies incremental, sparse and other variants, learning in graphical models, Springer (1998) 355–368.
M. Nimbarte and K. Bhoyar, Age invariant face recognition using convolutional neural network, Int. Journal of Elect. and Comp. Engineering8(4) (2018) 2126-2138.
S. Bianco, Large age-gap face verification by feature injection in deep networks, Pattern Recognition letters 90(2017) 36-42.
C. Nwankpa, W. Ijomah, A. Gachagan and S. Marshall, Activation functions: Comparison of trends in practice and research for deep learning, Machine Learning (2018) 1-20.
S. Hochreiter, The vanishing gradient problem during learning recurrent neural nets and problem solutions, Int. Journal of Uncert. Fuzz. and Knowledge-Based Sys. 6(2) (1998) 107–116.

S. Ioffe and C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, In: Proceedings of theInt. Conf. on Mach. Learning (2015) 448-456.
K. Messer, J.Matas, J. Kittler, J. Luettin and G. Maitre, XM2VTS: The extended M2VTS Database, In: Proceedings of the Int. Conf. on Audio and Video- based Biometric Person Authentication (1999)1-6.
D. Chen, X. Cao, F. Wen and J. Sun, Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification, In: IEEE Conf. on Comp. Visi.and Pattern Reco. (2013) 3025-3032.
H.V. Nguyen and L. Bai, Cosine similarity metric learning for face verification, In: Asian Conf. onComp.Visi. (ACCV) (2010) 709-720.
L. Wolf, T. Hassner and Y. Taigman, The one-shot similarity kernel, in: IEEE Conf. on Comp. Visi. (ICCV*)* (2009) 897–902.
D. Chen, X. Cao, L. Wang, F. Wen and J. Sun, Bayesian face revisited: A joint formulation, In: Euro. Conf. on Comp. Visi. (ECCV) (2012)566–579.
Q. Cao, Y. Yinga and P. Li, Similarity metric learning for face recognition, In: IEEE Conf. on Comp. Visi. (ICCV) (2013) 2408–2415.
D. Yi, Z. Lei, S. Liao and S. Z. Li, Learning face representation from scratch, arXiv preprint arXiv: 1411.7923, 2014.
Y. Sun, X. Wang and X. Tang, Deep learning face representation from predicting 10,000 classes, In: IEEE Conf. on Comp. Visi.and Pattern Reco. (2014)1891-1898.
M. M. Sawant, K. M. Bhurch, Age invariance face recognition: a survey on facial aging databases, techniques and effect of aging, Artificial Intelligence Review, Springer, 52 (2) (2019) 981-1008.
M. Heikkilä, M. Pietikäinen and C. Schmid, Description of interest regions with local binary patterns, Pattern Recognition42(3) (2009) 425–436.
B-C. Chen, C-S. Chen, W. H. Hsu, Cross-age reference coding for age-invariant face recognition and retrieval, In: Proceedings of the Euro. Conf. on Comp. Visi. (ECCV) (2014)768–783.
M. S. Shakeel and K.M. Lam, Deep-feature encoding-based discriminative model for age-invariant face recognition, Pattern Reco. 93(2019) 442-457.

Author Information

Ashutosh Dhamijaa
ECE department, SRM University, Sonepat, Haryana, India

R. B. Dubeyb
EEE department, SRM University,Sonepat, Haryana, India