Metaheuristic Approaches for Occlusion Invariant 3D Face Recognition Technique

A thesis

submitted in the fulfillment of the

requirement for the award of the degree of

Doctor of Philosophy

Submitted by

Sahil Sharma

Registration No. : 951503011

Computer Science and Engineering Department

Thapar Institute of Engineering and Technology, Patiala

INDIA-147004

AUGUST 2021

Certificate

I hereby declare that the work being presented in this thesis "Metaheuristic Approaches for Occlusion Invariant 3D Face Recognition Technique", in fulfillment of the requirements for the award of degree of **DOCTOR OF PHILOSOPHY** submitted in Computer Science and Engineering Department, Thapar Institute of Engineering and Technology, Patiala, is an authentic record of my work carried out under the supervision of Dr. Vijay Kumar, Assistant Professor, Computer Science and Engineering Department and refers other researcher works which are duly listed in the reference section. The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.

Sahil Sharma

(Sahil Sharma)
Registration No. 951503011

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge and belief.

Vijay

(Dr. Vijay Kumar)
Assistant Professor
Computer Science and Engineering Department
Supervisor

Acknowledgements

I owe to the grace of almighty, whose divine light provides me with the perseverance, guidance, patience, inspiration, faith, and strength to carry out this work.

I want to express my earnest gratitude to my supervisor, **Dr. Vijay Kumar,** Assistant Professor, Computer Science and Engineering Department, for his invaluable guidance and simulation throughout this research work, giving me constructive comments and support. I am incredibly indebted to him for providing time to listen and give his valuable suggestions. I am thankful to **Dr. Maninder Singh**, Professor and Head, Computer Science Department for being my administrative supervisor during my journey of Ph. D. This thesis was not possible without his support and cooperation.

I am grateful to the Doctoral committee comprising **Dr. Rinkle Rani**, Associate Professor, **Dr. Jhilik Bhattacharya**, Assistant Professor, Computer Science Department, and **Dr. Kulbir Singh**, Professor, Electronics and Communication Engineering Department, for monitoring the progress and providing valuable suggestions for the improvement of this work. I thank all the Computer Science and Engineering Department staff members and my colleagues **Dr. Ashish Girdhar** and **Dr. Simranjit Singh** for their support and cooperation. The discussions with them were motivating during this journey.

Words are inadequate in paying regards to my grandfather **Dr. Onkar Nath Sharma**, grandmother **Mrs. Sukhvarsha Sharma**, father **Mr. Suneet Sharma**, mother **Mrs. Sunita Sharma**, and my brother **Dr. Sushant Sharma** for their continuous encouragement and patience. I express deep gratitude to my wife **Ms. Ashima Sood**, whose love, care, and support have been an invaluable source of encouragement to me.

I would like to express thanks to all who contributed directly or indirectly during this journey of research work to make my Ph.D. a successful event.

Sahil Sharma

(**Sahil Sharma**)

Abstract

Face identification is one of the non - intrusive biometric strategies. Over the decades, several investigators have sought to achieve state-of-the-art face-recognition frameworks. Recently, AlexNet has proven to be the beginning of the deep learning age. Many face-recognition issues have been overcome in the last eight years by using deep learning methods ranging from multilayer perceptron to convolutional neural networks. Transfer learning methods have also been handy where the dataset is not huge, and we have a small number of subjects.

Researchers are focusing on 3D facial recognition and restoration for the last 3-4 years. There are various methods for 3D face reconstruction, namely, morphable model-based reconstruction, epipolar geometry-based reconstruction, and one-shot learning-based reconstruction, shape from shading-based reconstruction, as well as deep learning-based reconstruction.

3D face reconstruction techniques using voxels as well as facial landmarks have been proposed in this work. There are three face reconstruction techniques proposed in this work viz. 3D voxel-based face reconstruction using sequential deep learning, 3D landmarks-based face reconstruction, and voxel-based occlusion-invariant face recognition using game theory and simulated annealing. The three datasets, namely, Bosphorus Database, University of Milano Bicocca 3D Face Database, and Kinect Face Database, have been used for training and testing phases.

Using the game theory and simulated annealing method, the overall classification accuracy obtained from voxel-based facial recognition is 86.1%. For occlusion invariant facial recognition, the average accuracy generated by the proposed technique is 75.5%. In facial recognition using 3D landmarks, the average accuracy achieved from the given methodology is 81.3%. In the case of 3D mesh face recognition, the average precision achieved from the methodology is 83.9%. Adversarial generation technique for triplet generation promotes minimal bias. The technique coupled with simulated annealing allows the proposed system to be resilient in a variety of areas using voxels.

In the 3D voxel-based face reconstruction (3DVFR) method in the performance

evaluation phase, all predictions are shown with and without reconstruction. In terms of gender identification, the accuracy is greater than 90% following restoration for all three datasets due to binary classification. In emotional identification, the accuracy for the Bosphorus dataset is 94.57% after restoration compared to 83.95% before restoration, i.e., 10.62% improvement in accuracy. In the case of University of Milano Bicocca 3D face database and Kinect Face database, there is a rise of 14.9% and 12.49%. In occlusion classification, there is a rise of 4.07%, 4.82%, and 7.05% for Bosphorus, University of Milano Bicocca 3D face database, and Kinect Face databases. The precision of the proposed architecture (3DVFR) is state-of-the-art with 90.01% in 3D face recognition using Bosphorus dataset voxels. In the case of University of Milano Bicocca 3D face database and Kinect Face database, the facial recognition performance is 78.21% and 85.68%, respectively.

3D landmark-based facial reconstruction method has been compared with three recently developed approaches over three well-known databases. The experimental findings indicate that the proposed method obtained an 8-10 % increase over the current techniques. The computation period derived from the proposed method is roughly 50% quicker than the other approaches. Furthermore, in ablation experiments, the efficiency of the proposed model is checked for occluded versus non-occluded faces, gender, emotion, and occlusion type recognition.

Abbreviations

Adam	Adaptive Moment Optimizer
AE	Autoencoders
AUC	Area Under the Curve
CE	Cross Entropy Loss
CMC	Cumulative Matching Characteristic
CNN	Convolutional Neural Network
EPI	Epipolar Plane Images
FNR	False Negative Ratio
FPR	False Positive Ratio
GAN	Generative Adversarial Networks
GPU	Graphical Processing Unit
ICP	Iterative Closest Point
IoU	Intersection over Union
LPP	Locality Preserving Projections
LSTM	Long Short-Term Memory Network
MAE	Mean Absolute Error
MSE	Mean Square Error
NME	Normalized Mean Error
PCA	Principal Component Analysis
PSNR	Peak Signal to Noise Ratio
RL	Reinforcement Learning
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
ROC	Receiver Operator Characteristic
SA	Simulated Annealing
SGD	Stochastic Gradient Descent
SVM	Support Vector Machines
UMBDB	University of Milano Bicocca 3D Face Database
VAE	Variational Autoencoders

Notations and Symbols

TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
A,P,N	Anchor, Positive, Negative Triplet
i_t	Input Gate
f_t	Forget Gate
o_t	Output Gate
c_t	Current Memory Cell State

Contents

List of Figures

List of Tables

1. Introduction

1.1 Preamble

3D face recognition is widely used worldwide due to easily collectible 3D data and economical graphical processing units (GPUs). However, acquiring 3D images is harder as compared to 2D scans. Therefore, the number of images is limited in public databases [1]–[3].

Facial characteristics play a vital role when it comes to the recognition of humans with intelligent machines. 2D and depth image-based facial recognition cannot handle the challenges such as poor lighting, video monitoring, occlusion-invariant systems, etc. [4], [5]. 3D image segmentation is commonly used in the medical field [6]. The rapid evolution of 3D data capture devices has started producing a humongous number of 3D face datasets [7]. As a result, a substantial amount of detail is present in a single 3D image.

RGB-D image, point cloud image, and 3D mesh image are the various types of 3D faces available in the literature [8]. There are various challenges in 3D face recognition, viz. pose, occlusion, expression, lighting, etc. These variations affect the intra-class recognition capabilities of the 3D face recognition system [9]. Occlusion is one of the major challenges in 3D face recognition. There are mainly two occlusion types: internal (beard or pose variation) and external (glass, bottle, clothe, paper). 3D face restoration plays a pivotal role in occlusion handling. Clustering-based methods are useful in handling 3D data [10]. Restoration of 3D face utilizes point cloud, mesh, or depth information according to the data at disposal.

3D face recognition is used in the entertainment industry viz. 3D animation, augmented reality, and virtual reality. Advancements in 3D deep learning have enabled researchers to achieve 3D fashion modelling [11]. Deep learning has multiple applications [12], [13]. It is possible to build animated characters through Generative Adversarial Networks (GANs) [14], [15]. Ensembling, CNN, and transfer learning are some techniques that are commonly used by many researchers [16]–[18]. Various approaches have been proposed to address the problems associated with 3D facial restoration using facial geometry [19], [20]. Multi-modal autoencoders are used for representation learning [21]. The shape from shading (SFS) method uses the shades on the face because of lightening. SFS is profoundly sensitive to lightening and does not perform well under

pose variation. 3D morphable models (3DMM) are unable to generate accurate face shape [22]–[24]. Neural networks are used for attention-based signature verification [25].

1.2 General Framework of 3D Face Reconstruction

Training and testing are the two main phases of deep learning-based 3D face recognition (see Figure 1.1). There are two sub-phases in the training phase, namely, pre-processing and deep learning. During pre-processing, 3D face acquisition is made by RGB-D depth image, 3D face mesh image, and 3D point cloud image. Once the 3D face is acquired, face alignment and registration are done for maximum utilization of the available information. There are three methods for 3D face alignment. First, coarse-based facial landmark detection is one of the fastest methods. However, facial landmarks did not preserve the finer details of the face. To overcome this, voxelization is used that includes fine details of a 3D face. Voxelization is a slower process in contrast to landmarks detection. The third method uses a 3D object as a whole, in a 3D RGB-D image, mesh image, or the point cloud image.

After the facial alignment, the deep learning model can be trained. There are different types of deep learning models such as convolutional neural network (CNN), recurrent neural network (RNN), autoencoder (AE), generative adversarial network (GAN), and reinforcement learning (RL) [26], [27]. CNN and RNN are used in supervised learning for images and text, respectively. AE and GAN are used in semi-supervised learning. RL is used in unsupervised learning. The image segmentation-based method is implemented using morphological reconstruction [28]. Emotion recognition is done from face recognition under a virtual learning environment through a neural network [29]. Signature verification is done using a support vector machine and the neural classifier [30].

1.2.1 Training Phase

1.2.1.1 Voxelization

Voxel representation is widely used in multiple fields, viz. computer graphics, computational science, real-time computer vision, and 3D shape matching. Dynamic modeling requires voxelization or real-time scan conversion. In this process, the triangular mesh creates a voxel representation from the input surface [31].

Let point O be arbitrary origin point, and G be a polyhedron with triangular faces $t_1, t_2, t_3, ..., t_n$, then $H = \{H_1, H_2, H_3, ..., H_n\}$ be covering of G with 3D tetrahedra. H_i is defined by O and triangular faces t_i [32]. Point A is considered to be inside the polyhedron G iff

$$\sum_i \text{sign}(H_i)\ \text{incl}(H_i, A) > 0 \tag{1.1}$$

where sign(H_i) is true for $H_i > 0$, and incl(H_i, A) is true for all $A \in H_i$.

1.2.1.2 Locality Preserving Projections

Algorithm 1.1*: Locality Preserving Projections*

Input: Graph G with P Nodes

Output: l-dimensional vector

1. *Nodes a and b are connected iff*
   a) *Two nodes are close according to Euclidean norm R^n.*
      $\| x_a - x_b \|^2 < \in$ // $\in$ is number of neighbours
   b) *Nodes a and b are the k-nearest neighbors of each other.*
2. *Weighting of edges*
   a) *In case of given parameter p, $p \in R$, if nodes a and b are connected,*
      $W_{ab} = e^{-\frac{\|x_a - x_b\|^2}{p}}$
   b) *In case of a and b are connected by only one edge,*
      $W_{ab} = 1$
3. *Computation of eigenvalues and eigenvectors for generalized eigenvector problem*
   a) $ZL_m Z^T a = \lambda ZMZ^T a$

      Where M is the diagonal matrix with column sums of W. L_m is the Laplacian matrix, $L_M = M - W$. Z_i is the i_{th} column of matrix Z.

   b) *Suppose $d_1, d_2, ..., d_{l-1}$ be the column vector solution of Eq. 3(a), the ordered corresponding to eigenvalues $\lambda_0, \lambda_1, ..., \lambda_{\lambda-1}$. The final embedding is as follows.*

      $x_i \rightarrow y_i = S^T z_i$, $S = (S_0, S_1, ..., S_{l-1})$

      where y_i is l-dimensional vector, and S is n x l matrix.

Algorithm 1.1 depicts the locality preserving projections (LPP). Principal component analysis (PCA) [33] and LPP are two-dimensionality reduction algorithms. Clustering can also be used for reducing features [34]. Suppose there are large *n*-dimensional vectors of data points. The intrinsic property of data is used for dimensionality reduction of these large vectors. Locality preserving projection (LPP) builds a graph that consists of the neighborhood information of the dataset [35]

and solves the linear dimensionality reduction problem. LPP is a linear approximation of non-linear Laplace Eigenmap and is as follows [35], [36].

In the LPP algorithm, Graph G with P nodes are connected according to the Euclidean norm and k-nearest neighbors of two nodes (see Step 1). In Step 2, weights are assigned to nodes. In the final step, the final l-dimensional vector is computed based on the generalized eigenvector problem.

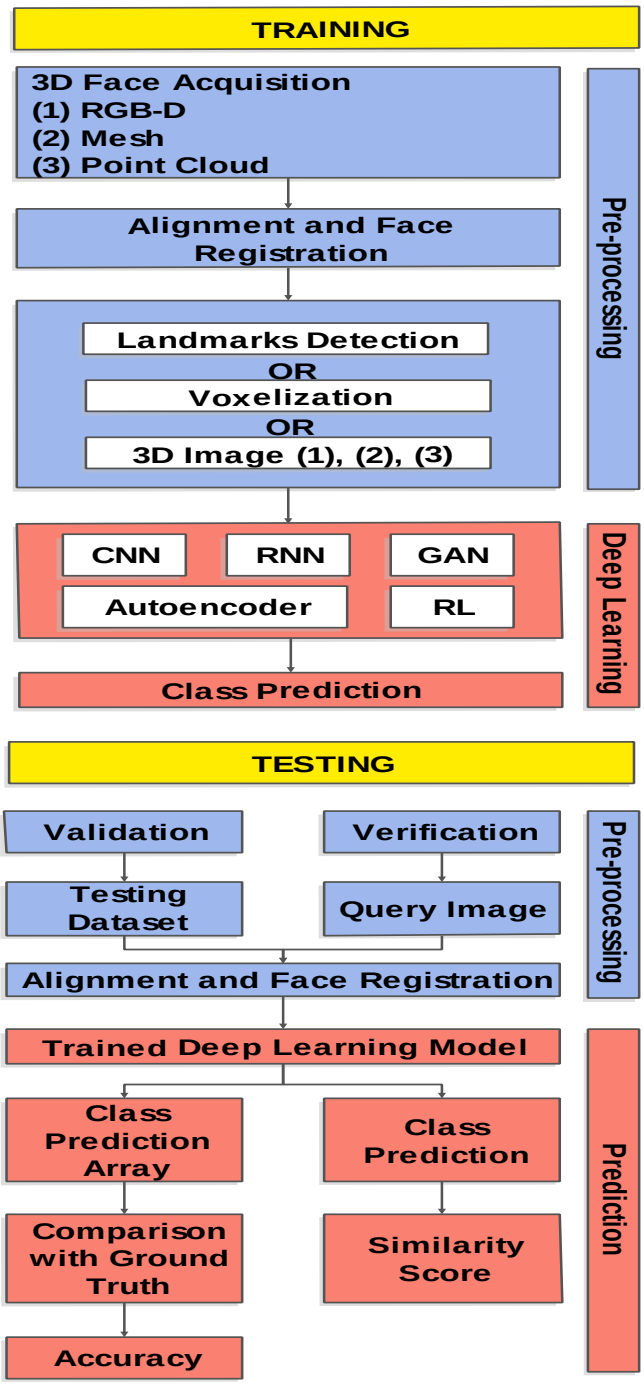


Figure 1.1 A general framework of deep learning-based 3D face recognition

1.2.1.3 Triplet Loss

Triplet loss uses face embeddings as vectors. It chooses three embeddings, namely, Anchor (A), Positive (P), and Negative (N), from the dataset such that A and P belong to the same class, and N belongs to a different class. A, P, and N are selected randomly based on three categories: easy triplets, hard triplets, and semi-hard triplets. Easy triplets (see Eq. 1.2) have a loss of 0. In hard triplets (see Eq. 1.3), negative embedding is closer to anchor embedding than positive embedding. In semi-hard triplets (see Eq. 1.4), the negative embedding is not closer than the positive but still has positive loss [37].

$$d(A, P) + margin < d(A, N) \quad (1.2)$$

$$d(A, N) < d(A, P) \quad (1.3)$$

$$d(A, P) < d(A, N) < d(A, P) + margin \quad (1.4)$$

The loss of a triplet (A, P, N) is defined as [19]:

$$L = max(d(A,P) - d(A,N) + margin, 0) \quad (1.5)$$

The main objective of triplet loss training is to find the global optima for the loss by pushing d(A, P) $\rightarrow$ 0 and d(A, N) > d(A, P) + margin by triplet loss training. Figure 1.2 represents the concept of triplet loss using three images in form of A, P, and N for triplet embeddings and triplet loss training.

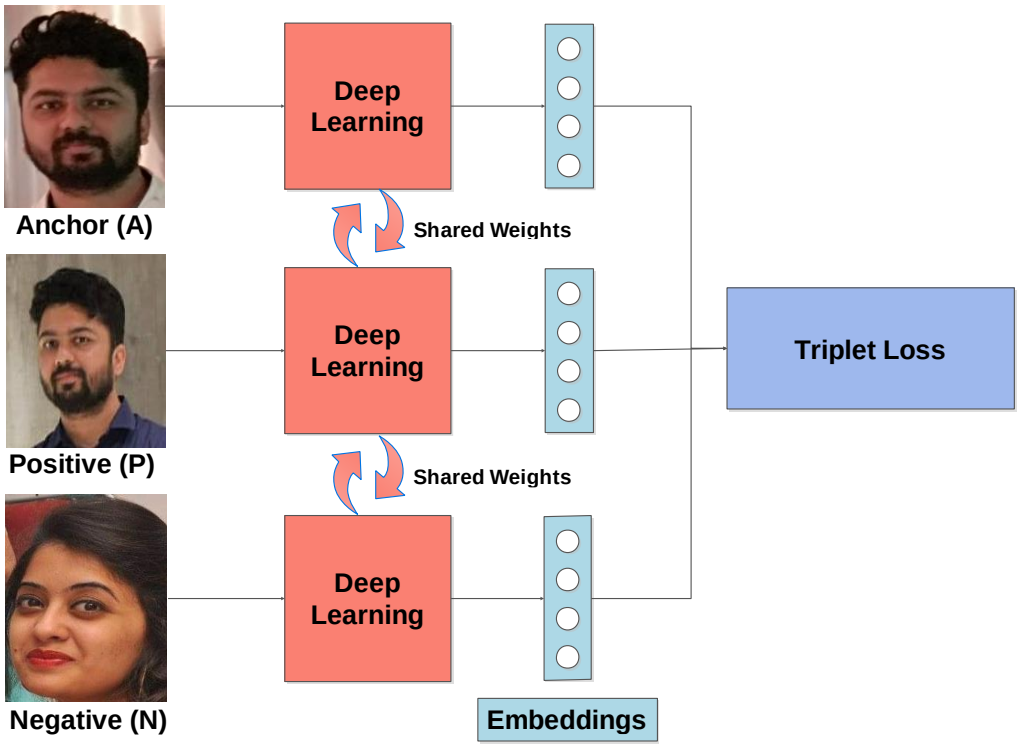


Figure 1.2 Concept of Triplet loss technique

1.2.1.4 Game Theory

Game theory is a term used jointly with generative adversarial networks (GANs). GANs [38], [39] are one of the types of generative models. They can be used to synthesize speech recognition [40], [41]. Let $P_{data}(I)$ be the distribution of a real image I, and $P_J(J)$ be the distribution of the input. Let generator G(z) capture the P_{data} distribution by using an adversarial process. The discriminator D distinguishes between real images and generated images. The formulation of the adversarial process in the form of a minimax game (see Eq. 1.6).

$$\min_G \max_D E_{I\sim P_{data}}[\log D(I) + \mathrm{E}_{Z\sim P_Z}[\log(1-D(G(z)))] \tag{1.6}$$

Theoretically, the global optimum $PG(Z) = P_{data}$ [42] is reached based on the minimax game on reaching Nash equilibrium [43] by the adversarial process. Recently, AttGAN [44] has achieved facial attribute editing as well as gender and age transformation (see Figure 1.3(A) and 1.3 (B)) viz. reconstruction of blond hair, eyeglass, changing expression, makeup, etc. Soft computing techniques are usable in finding global optimum [45]. Face aging with conditional GANs [46] achieved remarkable results in generating faces of different ages from a single image (see Figure 1.3 (C)).

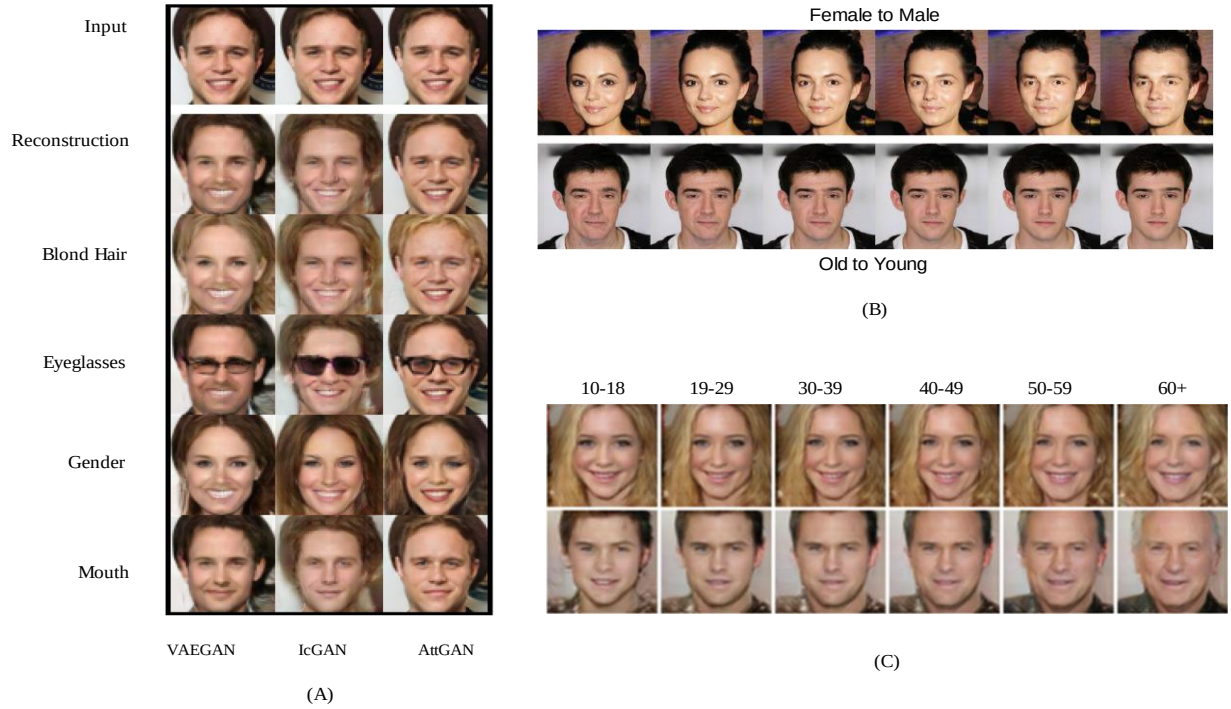


Figure 1.3 (A) Comparison of transforming attributes using VAE-GAN [47], IcGAN [48], and AttGAN [44], (B) Gender and age transformation using AttGAN, and (C) Transformation of face using Age Conditional GAN [46]

1.2.1.5 Simulated Annealing

The simulated annealing (SA) optimization algorithm is based on metallurgical practices. Particular material is heated at high temperature, and then it is brought to low temperature gradually. The shifting of atoms becomes unpredictable when heating a material at a high temperature. It helps in the elimination of impurities as the material takes pure crystal form after cooling. In terms of optimization, SA introduces a degree of randomness, which may take the solution from better to worse in an attempt to escape local minima and increasing the probability of achieving global optimum [49]. The applications of SA are diverse [50]–[52] by single criterion optimization [53].

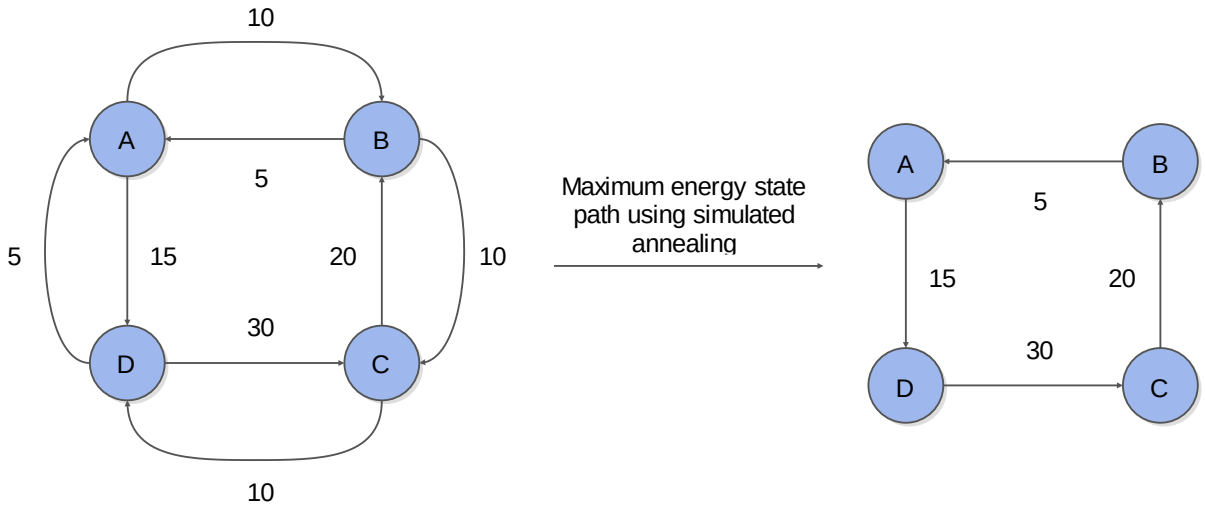


Figure 1.4 State energy maximization using simulated annealing

Figure 1.4 holds four states A, B, C, and D, having different energy. The main target is finding a path with maximum energy by using a simulated annealing algorithm to traverse every state exactly once. For understanding purposes, the four states have been connected clockwise and anticlockwise. The total sum of energies by clockwise traversing of all states is 35. In contrast, the total sum of energies by anticlockwise traversing is 70. Hence, the maximum energy state path is selected by anticlockwise traversing.

Algorithm 1.2*: Simulated Annealing*

Input: Random Initial state

Output: Path covering all states with maximum energy

1. $S = S_{init}$ *//Selection of initial state randomly*

2. *For $T = T_{max}$ to T_{min} do* *//New temperature in every iteration*
   a. $E_S = E(S)$ *//Current state energy*
   b. $N = next(S)$ *//New state*
   c. $E_N = E(N)$ *//New state energy*
   d. $\Delta E = E_N - E_S$ *//Change in energy*
   e. *If($\Delta E > \Theta$)* *//Maximization problem*
   f. $S = N$ *//New state selected*
   g. *Else if ($e^{\Delta E/T} > rand(0,1)$)*
   h. $S = N$ *//New state with probability $e^{\Delta E/T}$*
   i. *End if*

End for

The simulated annealing is shown in Algorithm 1.2. In a simulated annealing algorithm, the initial state is chosen randomly (see Step 1). E_s acts as current state energy. The new state becomes the current state if there is a positive energy change. Otherwise, the new state is chosen with probability $e^{\Delta E/T}$ (see Step 2). During the process, temperature T is decreased gradually so that solution converges towards the global optimum.

1.2.1.6 Variational Autoencoders

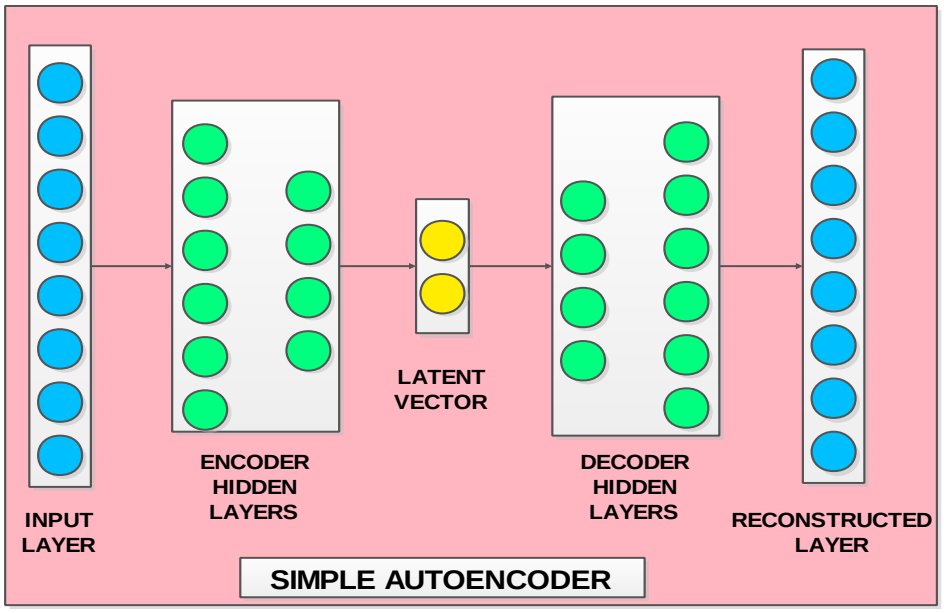
Variational autoencoders (*VAEs*) [54] are based on generative modeling to learn the variation in disentangled representations. Input data (x) is regenerated over the learned latent factors $z \in R^m$ by non-linear combination. For a given x, the probable posterior configurations of the latent factors (z) can be defined by a probability distribution $q_\phi(z|x)$ to maximize the probability of the observed data x on average over all samples that are possible from the latent transformation factors (z). This refers to the problem of optimization.

$$\max_{\phi,\theta} \; E_{q_\phi(z|x)}[\log \; p_\phi(x \mid z)] \tag{1.7}$$

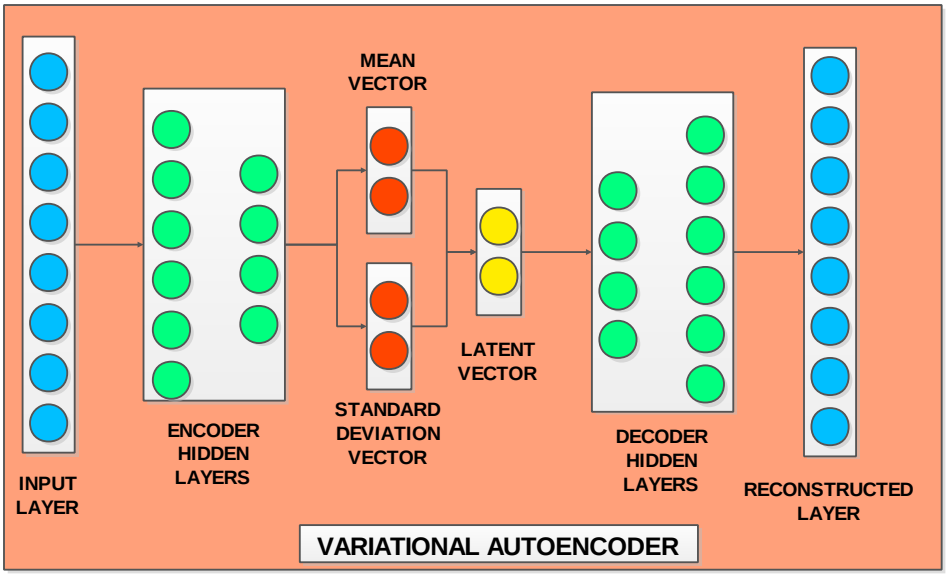
In studying disentangled representations of generative variables, a restriction is added that allows the distribution over latent factors z to be similar to independence prior and redundancy reduction. This results in a restricted optimization problem.

$$\max_{\phi,\theta} \; E_{q_\phi(z|x)}[\log \; p_\phi(x \mid z)] \text{ subject to } D_{KL}(q_\phi(z \mid x) \mid p(z)) < \varepsilon \tag{1.8}$$

where ε is the strength of constraint.



(a)



(b)

Figure 1.5 Architectural view (a) simple autoencoder (b) variational autoencoder

Kullback-Leibler divergence (D_{KL}) [55] measures the difference between two probability distributions. Eq. 1.8 can be expressed in lagrangian form. The traditional variational free energy goal function can be achieved.

$$L(\theta,\phi;x) = E_{q_\phi(z|x)}[\log p_\theta(x|z)] - \beta D_{KL}(q_\phi(z|x) \| p(z)) \quad (1.9)$$

where $\beta \geq 0$ denotes the regularization constraint.

The deep unsupervised models learn the disentangled latent space (z) by using constrained optimization. β coefficient is the hyperparameter of the VAE model. β is a coefficient to balance the prior matching costs and magnitude of gradients from the restoration during VAE encoder training. Figure 1.5 presents the architectural view of a variational autoencoder and simple autoencoder.

1.2.1.7 Bidirectional Long Short-Term Memory

Bidirectional Long Short-Term Memory (BiLSTM) [56] layer allows the network to learn long-term dependencies between time instances of embedding during forward and backward propagation. The cell state and the hidden state are part of the layer. The BiLSTM layer is part of the hidden state depending upon forward or backward propagation. Information learned from the previous time steps is part of the cell state. The BiLSTM layer adds or removes the information from the cell state at every time step. Input gate (i_t), forget gate (f_t) and the output gate (o_t) controls these updates of the layer.

Figure 1.6 presents the essence of BiLSTM architecture. It leverages forward and backward propagation to learn the long-term dependencies, enabling us to classify the previous and future states of each element sequence-wise.

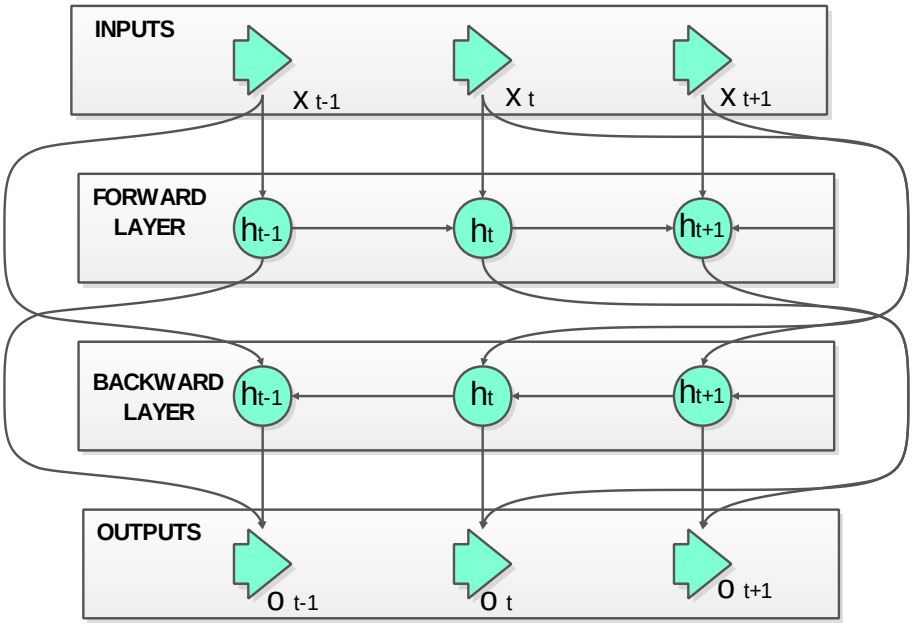


Figure 1.6 Architecture of BiLSTM

For different time steps, BiLSTM has various neural networks in series. The i_t, f_t, and o_t gates control the current time step (x_t) input and the cell state (c_t) which is defined as the old hidden state (h_{t-1}) for each time step. The current hidden state (h_t) and current memory cell (c_t) is updated using x_i, h_{t-1}, i_t, f_t, and o_t.

Table 1.1 shows the BiLSTM parameters that controls the cell state and hidden state of the layer.

Table 1.1. BiLSTM Parameters

Parameter	**Equations**
Input Gate	$i_t = \sigma_g (W_i \otimes [h_{t-1}, x_t] + b_i$
Output Gate	$o_t = \sigma_c (W_o \otimes [h_{t-1}, x_t] + b_o$
Forget Gate	$f_t = \sigma_g (W_f \otimes [h_{t-1}, x_t] + b_f$
Cell States	$c_t = f_t \otimes c_{t-1} + i_t \otimes q_t$
Cell Outputs	$h_t = o_t \otimes \sigma_c(c_t)$
Input Weights	W_i, W_o, W_f

In Table 1.1, σ_g is the sigmoid activation function. σ_c denotes the hyperbolic tangent activation function, and $\otimes$ represents the element-wise multiplication [56].

1.2.1.8 Support Vector Machine

The support vector machine (*SVM*) has been used extensively for face classification [57], [58]. The deep embedding calculated from the previous pipeline is trained on Gaussian radial basis kernel [59] for recognition accuracy.

In case of voxel-based face recognition, the accuracy is computed by using the ensemble of voting for 8^3, 16^3, and 32^3 voxels predictions. The ensemble is formally represented as:

$$Ensemble\ Class\ (C_E) = \text{mode}(8^3\ class,\ 16^3\ class,\ 32^3\ class) \quad (1.10)$$

Figure 1.7 illustrates the process of ensemble implemented after the implementation pipeline of deep learning layers.

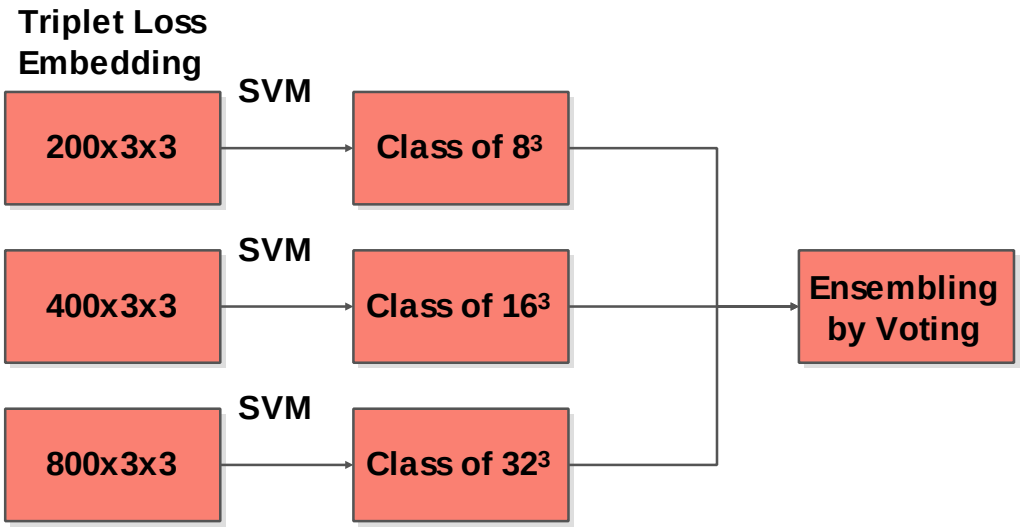


Figure 1.7 Ensembling of Voxels

1.2.2 Testing Phase

There are two sub-phases in the testing phase, namely, validation and verification.

1.2.2.1 Validation

The validation is implemented over the test dataset, where the ground truth values are provided. This helps in calculating the accuracy of the whole test dataset. This prediction is done using the trained classification model. The basic overview of validation is illustrated in Figure 1.8.

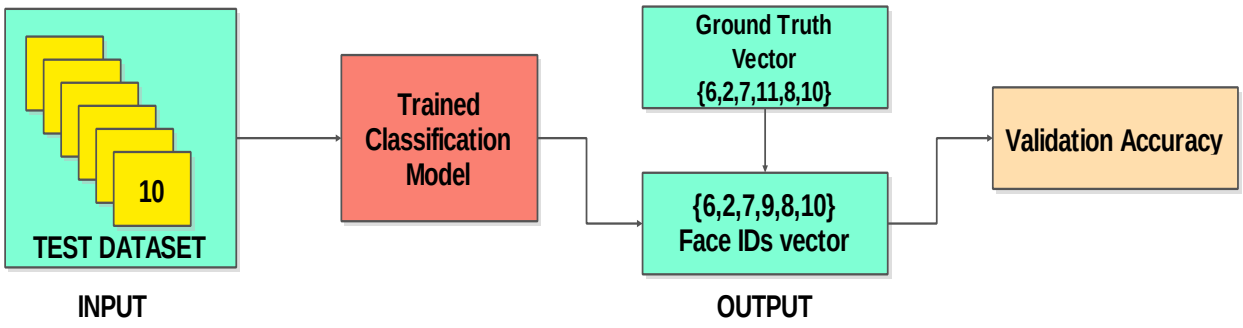


Figure 1.8 Process of validation

The overall accuracy is calculated on comparison of ground truth vector with the face IDs vector. Eq. 1.11 shows the equation of verifying the accuracy of the model.

$$Accuracy = \frac{Number\ of\ IDs\ matching}{Total\ number\ of\ IDs} * 100 \tag{1.11}$$

1.2.2.2 Verification

Verification means a comparison between the query image and the ground truth value. In Figure 1.9, an image is passed with 10 written in it. As it passes from the trained classification model, the binary value is calculated as the verification accuracy based on the match of ground truth value and the predicted class value.

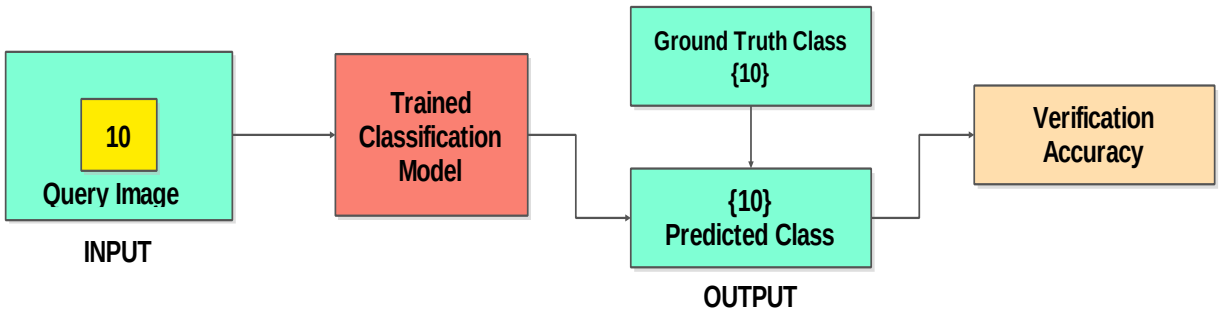


Figure 1.9 Process of verification

1.3 3D Face Reconstruction Techniques

In this section, the well-known 3D face reconstruction techniques are discussed.

1.3.1 3D Morphable Model-based Reconstruction

A 3D Morphable Model (3DMM) is a generative model for facial appearance and shape. There are two main ideas around it. Firstly, all the faces to be generated are in a dense point-to-point correspondence that can be achieved by doing the face registration using some face images. The morphological faces (*morphs*) are generated using this dense correspondence. Secondly, this technique focuses on disentangling the facial color and shape from the other factors, namely illumination, brightness, contrast, etc. [60]

3DMM was introduced by Blanz and Vetter [61]. In the literature, there are multiple variations of 3DMM in today's times [62]–[66]. Those models make low-dimensional representations for the face expression, texture, and identity using PCA with multiple facial scans. Basel Face Model

(BFM) is one of the publicly available 3DMM models. It constructs the model by registering the template mesh corresponding to the scanned face by applying Iterative Closest Point (ICP) and Principal Component Analysis (PCA) [67].

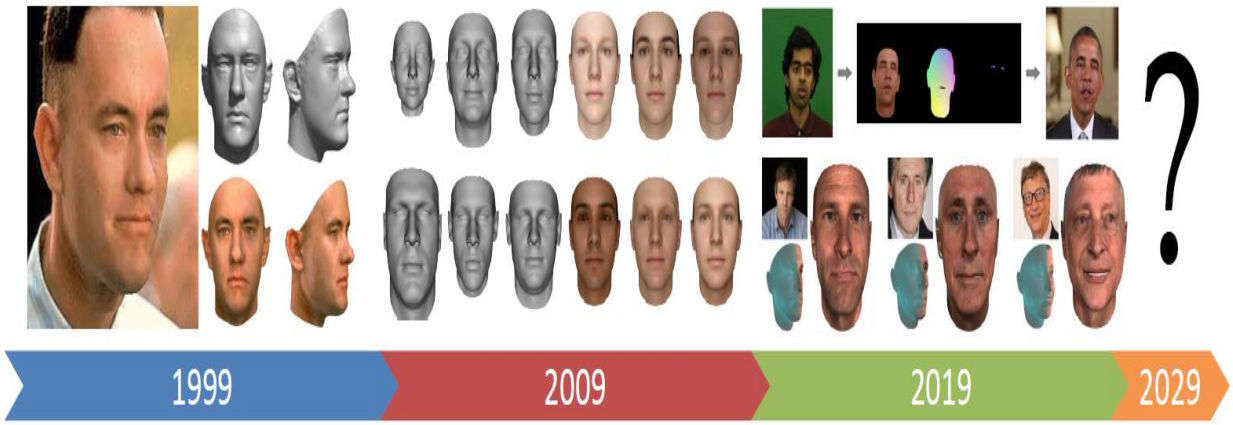


Figure 1.10 Progressive improvement in 3DMM in last twenty years

Figure 1.10 shows the different phases of 3DMM models. The progressive improvement in 3DMM has been taken from [9], [39], [61], [68].

1.3.2 Epipolar Geometry based Reconstruction

Epipolar geometry-based facial reconstruction approach uses various non-synthesizing perspective images of one subject to generate a single 3D image [20]. Good geometric fidelity is one of the advantages of this technique. It attains more accuracy as it utilizes the various images from different angles. The requirement of more than one calibrated camera and the orthogonal images are the two challenges of this technique. Figure 1.11 shows the horizontal and vertical Epipolar Plane Images (EPIs) obtained from the central view and the sub-aperture images [20].

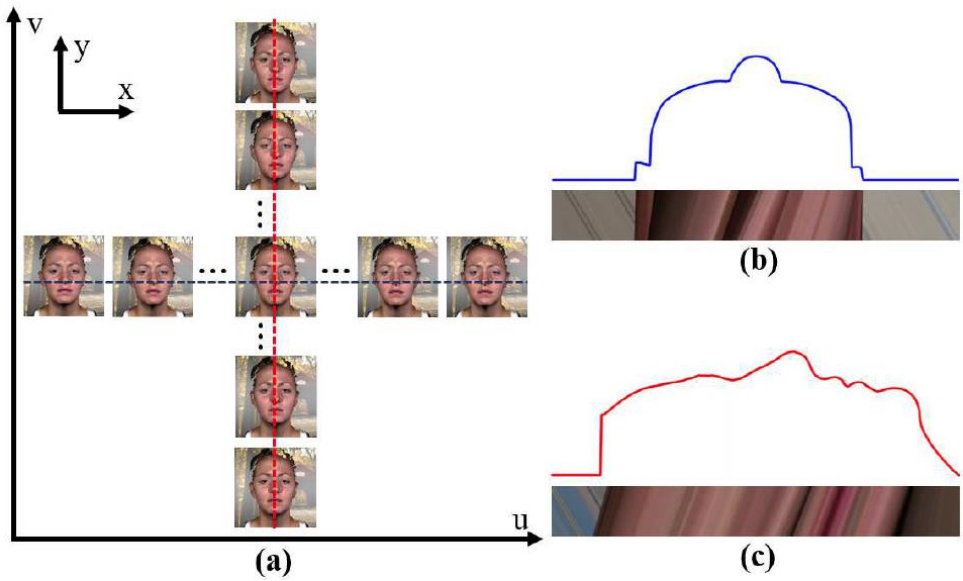


Figure 1.11 (a) Epipolar Plane Images corresponding to 3D face curves, (b) horizontal EPI, and (c) vertical EPI [20]

1.3.3 One Shot Learning based Reconstruction

One-shot learning is a method in which a single image of an individual is used to recreate the 3D recognition model [69]. The advantages are that a single image is required per subject to train the model. Hence, it is quicker to train and also generates promising results [70]. However, this approach cannot be generalized to videos. One-shot learning-based 3D reconstruction is an active research area. The ground truth 3D models are needed to train the model to map 2D-to-3D image. Some researchers have used depth prediction to reconstruct 3D structures [71], [72]. While other techniques directly predict 3D shapes [73], [74]. Few works have been done on 3D face reconstruction by utilizing one 2D image [22], [75]. The optimum parameter values for each 3D face are obtained using the deep neural networks and the model parameter vector. As a result, the major improvement has been achieved over [76], [77]. However, this approach fails to handle pose variation adequately. The major issues of this technique are the creation of multi-view 3D faces and reconstruction degradation. The general framework is shown in Figure 1.12.

1.3.4 Shape from Shading based Reconstruction

The shape from shading (SFS) method is based on the recovery of 3D shapes from shading and lighting cues [78]. It uses an image that produces a good shape model. However, occlusion cannot be dealt with when shape estimates are interfered with a shadow of the target. It only operates well

under lightening from the non-frontal face view (see Figure 1.13).

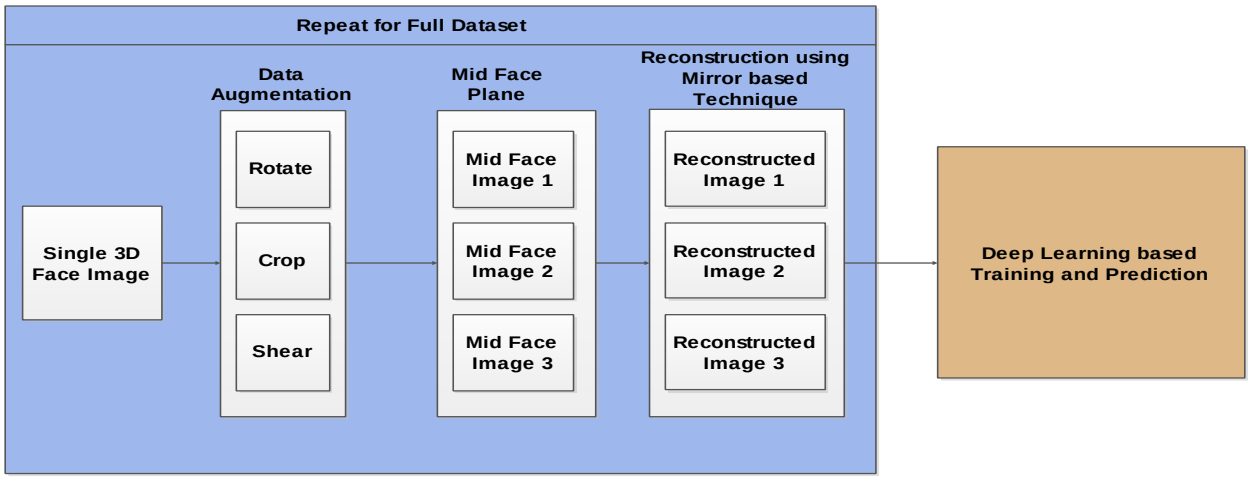
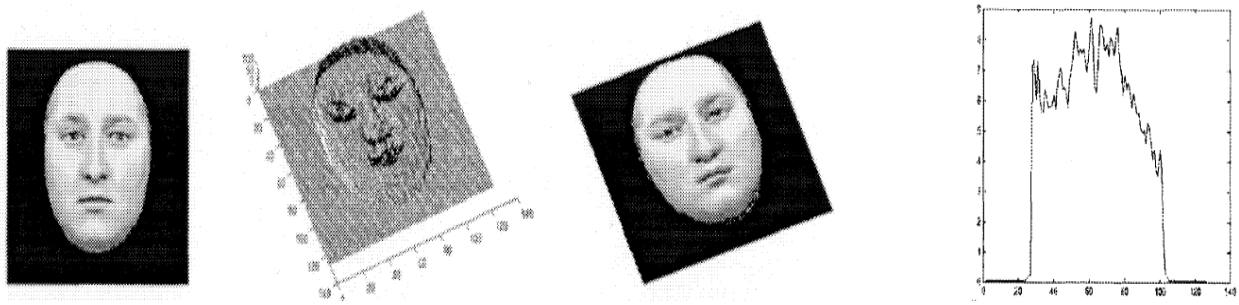


Figure 1.12 General framework of one shot learning based 3D face reconstruction



(a) (b) (c) (d)

Figure 1.13 3D face shape recovery (a) 2D image, (b) 3D depth image, (c) Texture projection, and (d) Albedo histogram [78]

1.3.5 Deep Learning based Reconstruction

The 3D generative adversarial networks (3DGANs) and the convolutional neural networks (3DCNN) are the deep learning methods around which the 3D face reconstruction methods are being built [39]. The advantages of this method are high fidelity and better performance. However, it takes a lot of time to train GANs. The face reconstruction in canonical view can be done by the face-identity preserving (FIP) method [79]. [80] introduced a multi-layer generative deep learning model for image generation under new lighting. In face recognition, the training corpus is responsible for providing the labels for training a multi-view perceptron-based approach. Synthetic data is augmented from a single image using facial geometry to train a CNN [22]. [75] proposed the unsupervised version of this reconstruction. Supervised CNNs are used to implement facial

animation tasks [81]. The 3D texture and shape are restored using the deep convolutional neural networks (DCNNs). 3DMM does the synthesis of images for depth estimation using an image-to-image translation-based deep convolutional neural network [82]. In [83], facial texture restoration has greater fine details than the 3DMM based estimated texture [84]. Figure 1.14 shows different phases for 3D face recognition by restoration of the occluded region.

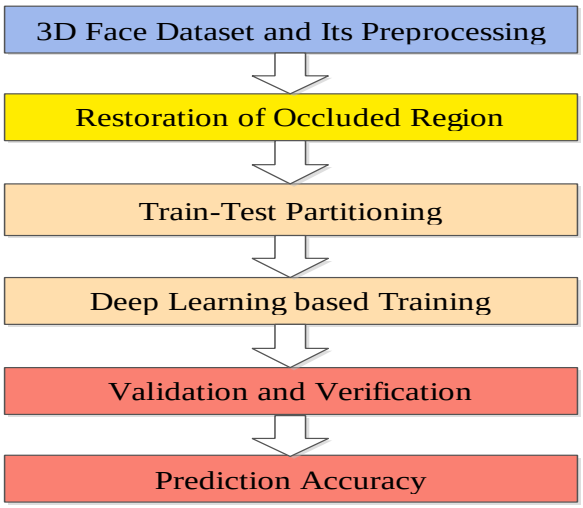


Figure 1.14 Phases of 3D face recognition using restoration

1.4 Performance Evaluation Metrics

Based on the confusion matrix (see Figure 1.15), the multiple performance evaluation metrics are computed.

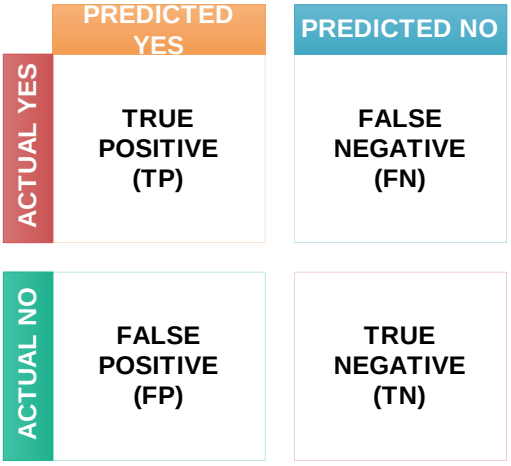


Figure 1.15 Confusion Matrix

It is important to understand the concepts of true positive (TP), true negative (TN), false-positive (FP), and false-negative (FN). When actual and predicted values are 'YES', it is known as TP.

When both values are ‘NO’, it is known as TN. In FN, the actual value is ‘YES’, but predicted is ‘NO’. For FP, the actual value is ‘NO’, but the predicted is ‘YES’.

1.4.1 Accuracy

The accuracy of the prediction model is calculated by checking the correctness of predictions in a specified task. If the data is biased, the accuracy gets affected. Synthetic data and data augmentation are the two majorly used techniques in eliminating biasness [85]. It can be defined as [86]:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{1.12}$$

1.4.2 Sensitivity

Sensitivity is the correct classification of all the true positives and is defined as [86].

$$Sensitivity = \frac{TP}{TP+FN} \tag{1.13}$$

1.4.3 Specificity

Specificity is the correct classification of all the true negatives and is defined as [86].

$$Specificity = \frac{TN}{TN+FP} \tag{1.14}$$

1.4.4 Precision

Precision is the ratio of actual positive values compared to total positive values, including the predicted ones. It is mathematically defined as:

$$Precision = \frac{TP}{TP+FP} \tag{1.15}$$

1.4.5 Recall

The recall is the correct classification of all the true positives and is defined as [86].

$$Recall = \frac{TP}{TP+FN} \tag{1.16}$$

1.4.6 F1 Score

F1 Score is the harmonic mean of precision and sensitivity value. It is defined as:

$$F1\ Score = \frac{2*TP}{2*TP+FP+FN} \tag{1.17}$$

The F1 score is useful in cases where there is a class imbalance.

1.4.7 Logarithmic Loss or Log Loss

Log Loss is the quantification of the classifier by penalizing the wrong classifications. It is heavily used in data science competitions on Kaggle. It is defined as:

$$LogLoss = -\frac{1}{N}\sum_{1}^{N} y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \tag{1.18}$$

where N is the number of samples, y_i is the actual value, and $\hat{y}_i$ is the predicted value.

1.4.8 False Positive Ratio

False Positive Ratio (FPR) is the ratio of wrongly predicted negative values to the total number of negative values in actual and predicted. FPR is defined as:

$$FPR = \frac{FP}{FP+TN} \tag{1.19}$$

1.4.9 False Negative Ratio

False Negative Ratio (FNR) is the ratio of wrongly predicted positive values to the total number of positive values in actual and predicted. The mathematical definition of FNR is as: [87]

$$FNR = \frac{FN}{FN+TP} \tag{1.20}$$

1.5 Loss Functions, Activation Functions, and Optimizers

In this section, the well-known classification loss functions, activation functions, and optimizers are mentioned.

1.5.1 Loss Functions

This sub-section discusses the classification loss functions.

1.5.1.1 Binary Cross-Entropy

The binary cross-entropy loss function is used in binary classification viz. gender recognition, presence of occlusion, verification of a person, etc. [88]

$L(Y, \hat{Y}) = -Y \log(\hat{Y})$, when $Y = 1$ (1.21)

$L(Y, \hat{Y}) = (Y - 1) \log(1 - \hat{Y})$, when $Y = 0$ (1.22)

Where Y is the actual value and $\hat{Y}$ is the predicted value.

1.5.1.2 Categorical Cross-Entropy

The categorical cross-entropy loss function is calculated using the one-hot encoding method for labeling the output values. The one-hot encodings have to be consistent with the training, validation, and testing dataset. For the Modified National Institute of Standards and Technology (MNIST) dataset, the classes 0-9 should have labels using one-hot encoding vectors as [88]

Label 0: [1,0,0,0,0,0,0,0,0,0]
Label 1: [0,1,0,0,0,0,0,0,0,0]
Label 2: [0,0,1,0,0,0,0,0,0,0]
.
.
Label 9: [0,0,0,0,0,0,0,0,0,1]

1.5.1.3 Sparse Categorical Cross-Entropy

The categorical cross-entropy loss function is calculated using the classes to be singular numbers[89].

Label 0: [0]
Label 1: [1]
Label 2: [2]
.
.
Label 9: [9]

1.5.2 Activation Functions

The activation functions are responsible for passing the information from one layer to another in a neural network. The activations must be computationally efficient as they are used whenever each neuron fires in a deep learning network. All the neural network-based models use different

activation functions while performing the various operations, viz. forward and backward propagation.

1.5.2.1 Sigmoid

The sigmoid activation function is used for binary classification. The threshold value for rounding off the output is generally 0.5 in the case of Sigmoid (see Figure 1.16a). The mathematical formulation of the sigmoid function is as follows:

$$Sigmoid(z) = \frac{1}{1+ e^{-z}} \tag{1.23}$$

where $z = wx + b$ with x as input vector, w as weight, and b as bias.

1.5.2.2 Hyperbolic Tangent ($tanh$)

The $tanh$ function is the non-linear activation function that can be used in place of the sigmoid. The threshold for $tanh$ is 0 (see Figure 1.16b). The mathematical representation of hyperbolic tangent is as follows [90]:

$$\tanh(z) = \frac{e^{z}- e^{-z}}{e^{z}+ e^{-z}} \tag{1.24}$$

The value of $\tanh(z)$ lies in the range of -1 and 1.

1.5.2.3 Rectified Linear Unit (ReLU)

Rectified Linear Unit (ReLU) is the widely used activation function in the research community. The rectified linear unit (ReLU) is the non-linear activation function, which does not bound the output to the upper value. ReLU activation function calculates the value of z by taking the maximum between 0 and z (see Figure 1.16c).

$$relu(z) = \max(0, z) \tag{1.25}$$

1.5.2.4 Leaky ReLU (*lrelu*)

Leaky ReLU (*lrelu*) does not make the negative component zero. However, it keeps the maximum value in the range of z and 0.1z. For example, if z = -100, using ReLU the activation value will be

0. Whereas using leaky ReLU, the activation value will be -10 (see Figure 1.16d).

$$lrelu(z) = \max(0.1z, z) \tag{1.26}$$

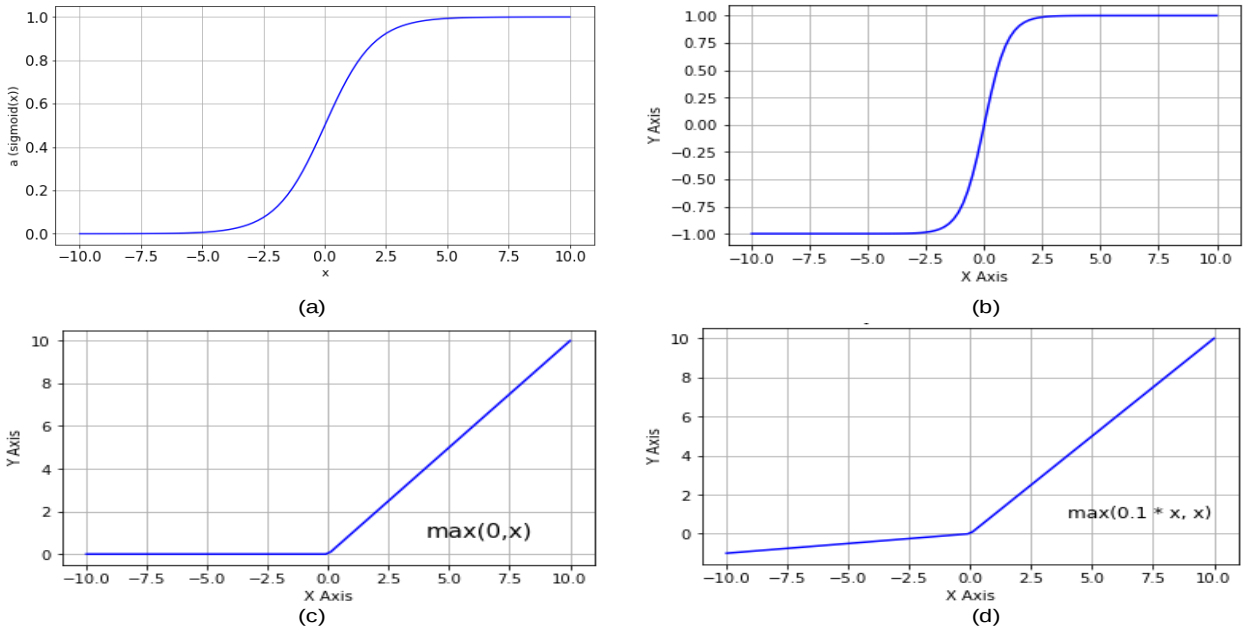


Figure 1.16 (a) Sigmoid activation function, (b) tanh activation function, (c) ReLU, and (d) leaky ReLU activation function

1.5.2.5 Softmax

The softmax activation function is helpful for multiclass classification. It maintains the array of probabilities for all the classes. The sum of all the probabilities in output is 1. The class has a maximum value of probability is the final prediction (see Figure 1.17).

$$Softmax\ (z) = \frac{e^{z}}{\sum_{i=1}^{i=n} e^{z_i}} \tag{1.27}$$

where n = number of samples, z^i is i^{th} neuron value in the layer

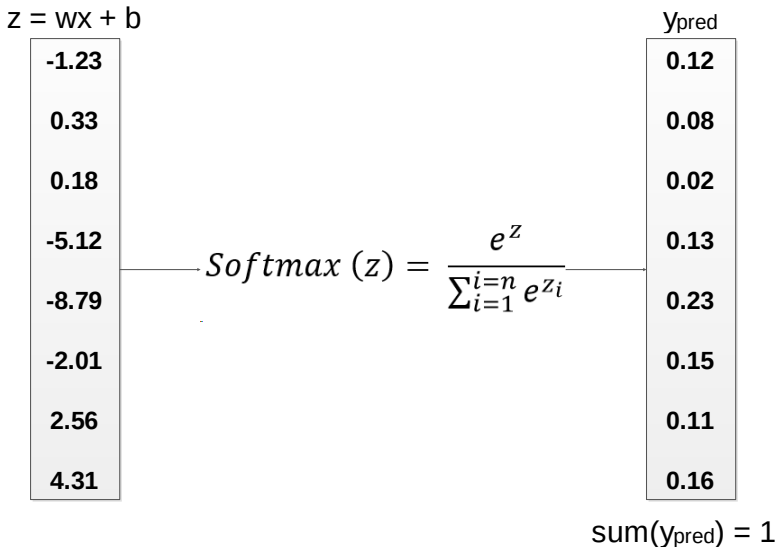


Figure 1.17 Softmax activation function

1.5.3 Optimizers

1.5.3.1 Stochastic Gradient Descent

Stochastic Gradient Descent (SGD) is an optimizing algorithm, which calculates the gradients on the subsets or batches in the deep learning process. The gradient descent algorithm calculates the value of the objective function at the end of every epoch. Whereas SGD gives the constant monitoring of the change of value for objective function with every mini-batch.

1.5.3.2 Root Mean Square Propagation

Root Mean Square Propagation (RMSProp) is an optimizing algorithm responsible for reducing the motion in the vertical direction and increasing the motion of gradient descent in the horizontal direction towards the global minima.

1.5.3.3 Adaptive Moment Estimation

Adaptive moment estimation (Adam) optimizer refers to an ensemble of gradient descent with momentum and RMSProp. There are multiple hyper-parameters, which are responsible for improving the performance of Adam optimizer viz. learning rate (α), momentum-based friction parameter (β_1), RMSprop based friction parameter (β_2), and the epsilon (ϵ) [91].

1.6 Application of 3D Face Recognition

1.6.1 Facial Puppetry

The virtual characters in the games and movie industry use facial cloning or puppetry in video-based facial animation. The expressions and emotions must be transferred from the user to the target character in input through video streaming.

Figure 1.18 Face puppetry demonstration in real-time by digital avatars [92], [93]

1.6.2 Speech-driven Animation and Re-enactment

Zollhofer et al. [94] discussed various video-based face reenactment works. Most of the methods in the literature depend on the reconstruction of source and target face using a parametric facial model.

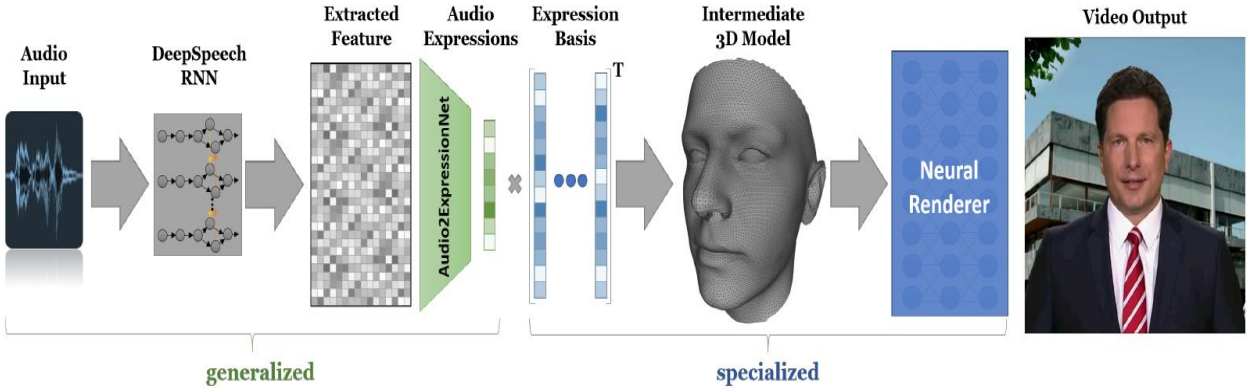


Figure 1.19 The Pipeline of Neural Voice Puppetry [95]

1.6.3 Video Dubbing

Dubbing is an important part of filmmaking where an audio track is added or replaced in the original scene. The original actor's voice is to be replaced with a dubbed actor. This process requires ample training for dubbed actors to match their audio with the original actor's lip-sync. To minimize the discrepancies in visual dubbing, the reconstruction of mouth in motion is done to complement it with the dialogues spoken by a dubbed actor. It also involves the mapping of dubber's mouth movements with the actor's mouth. Hence, the technique of image swapping or transferring parameters is used.

Figure 1.20 Visual dubbing by VDub [96] and Face2Face with enabled live dubbing [15]

1.6.4 Virtual Make-Up

Virtual makeup will be of excessive use in online platforms for meetings and video chats where a presentable appearance is indispensable. It includes the changes in the digital image such as applying suitable colour lipstick, face masks, etc. It can be useful for beauty product companies as they can advertise digitally, and consumers can experience the real-time effect of their products on their images. It is implemented by using different reconstruction algorithms (see Figure 1.21).

1.6.5 Projection Mapping

As the name suggests, projection mapping uses the projectors to amend the features or expressions of real-world images. This technique is used to bring life in static images and given them visual display. Different methods are used for projection mapping in 2D and 3D images, such as different

projectors, to alter the appearance of a person (see Figure 1.22).

Figure 1.21 Synthesized virtual tattoos adjusting to facial expression [97]

Figure 1.22 FaceForge based live projection mapping [98]

1.6.6 Face Replacement

Face Replacement is commonly used in the entertainment industry where the source face is to be replaced with the target face. This technique is based on parameters such as the track of identity, facial properties, and expressions of both faces (source and target). The source face is to be rendered so that it matches the conditions of the target face.

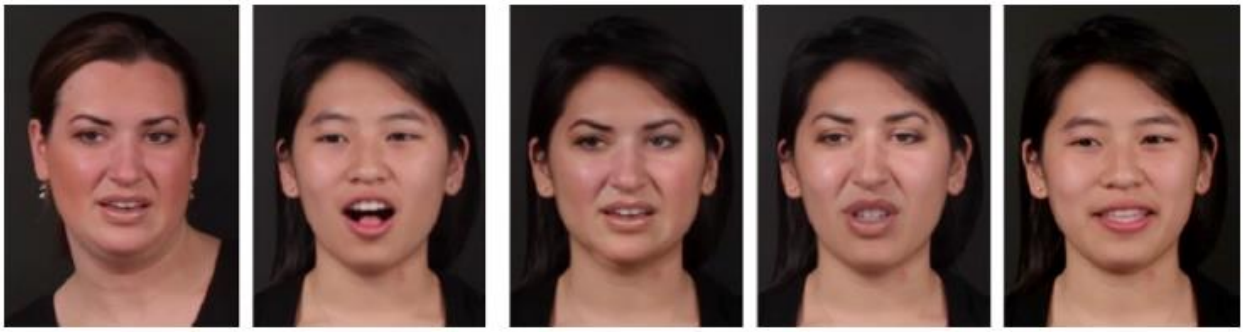


Figure 1.23 Expression invariant face replacement system

1.6.7 Face Aging

Face aging is an effective technique to convert 3D face images into 4D. If a single 3D image can be synthesized using aging GAN, then it would be useful in creating 4D datasets. Face aging is also called as age progression or age synthesis as it revives the face by changing the facial features. Various techniques are used to enhance the features of the face so that the original image is preserved.

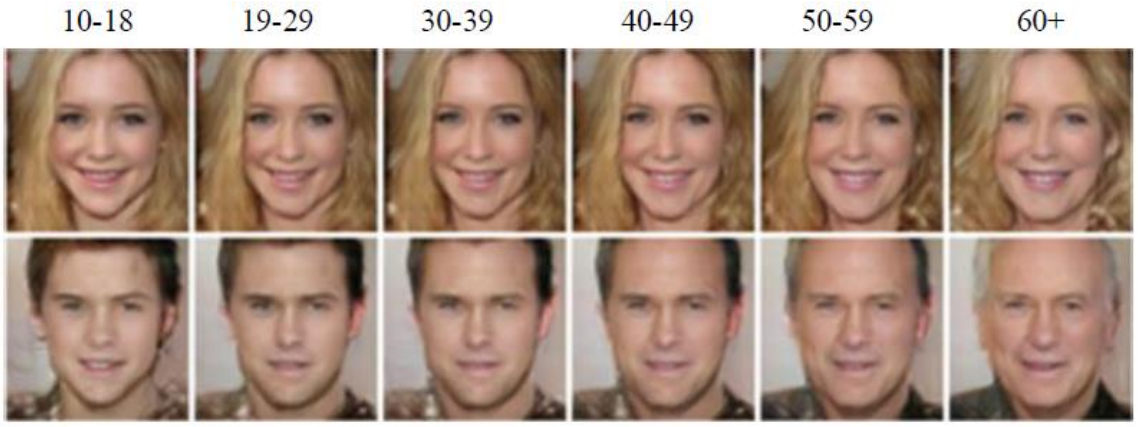


Figure 1.24 Transformation of face using ACGAN [46]

1.7 Research Motivation

There are many areas in 3D face reconstruction on which the practical work has to be done. Due to the advent of deep learning techniques and GPUs availabilities on the cloud, such as Google Cloud Platform (GCP), Microsoft Azure, Amazon Web Services (AWS), and the Google Colaboratory, the research work has become easier.

Considering the deep learning advancements in recent years, 3D face recognition and

reconstruction have become the highlight in face recognition. The researchers are developing novel algorithms and applications for 3D face reconstruction. Due to the COVID-19 pandemic, video calls have become the norm. Anyone can take advantage of deep learning-based 3D face make-up applications during online meetings.

These facts motivated me to develop new reconstruction algorithms and applied on face recognition techniques.

1.8 Thesis Organization

This thesis is devoted to the development of 3D face reconstruction techniques. The review of existing techniques is mentioned in Chapter 2. Chapter 3 presents the 3D face recognition framework using the concepts of game theory and simulated annealing. Chapter 4 presents the 3D face reconstruction technique for voxel-based images. Chapter 5 presents the 3D landmarks-based face restoration technique.

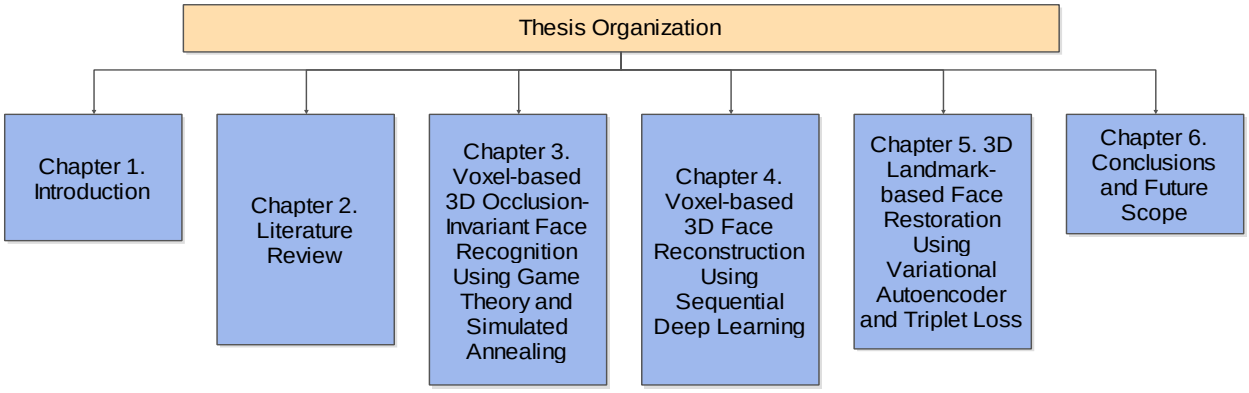


Figure 1.25 Thesis Organization

Chapter 1: Introduction

This chapter discusses the foundation of important concepts of 3D face recognition technique. The general framework of 3D face reconstruction along with mathematical formulation is studied. Five well-known 3D face reconstruction techniques are discussed. This chapter discusses the performance evaluation metrics and applications of 3D face recognition system.

Chapter 2: Literature Review

In chapter 2, various 3D face reconstruction techniques are presented in the literature. 3D morphable model-based reconstruction, epipolar based face reconstruction, one-shot learning-based reconstruction, and deep learning-based face reconstruction are discussed with their pros

and cons. Research gaps and objectives are also discussed.

Chapter 3: Voxel-based 3D Occlusion-Invariant Face Recognition Using Game Theory and Simulated Annealing

In Chapter 3, the proposed face recognition framework is discussed in detail along with its computational complexity. The preliminary concepts of voxelization, locality preserving projections, triplet loss, game theory, and simulated annealing are discussed. The main objective of this chapter is to present the occlusion-invariant 3D face recognition technique after reducing the triplet loss below the threshold value using a discriminator based on the concept of simulated annealing.

Chapter 4: Voxel-based 3D Face Reconstruction Using Sequential Deep Learning

In chapter 4, the proposed 3D face recognition framework is designed to recognize a person's gender, emotion, occlusion, and face. A series of deep learning models, including variational autoencoder, BiLSTM, and triplet loss training, are applied to generate deep learning-based embeddings. After that, the embeddings are passed through a support vector machine to predict the final class using voting-based ensembling. The main objective of this chapter is to present the voxel-based face reconstruction technique using epipolar geometry. The ablation studies show that the 3D face recognition after reconstruction achieves state-of-the-art results for occlusion, emotion, gender, and person identification.

Chapter 5: 3D Landmark-based Face Restoration Using Variational Autoencoder and Triplet Loss

In chapter 5, a novel 3D face recognition framework is proposed using variational autoencoder and triplet loss. Under the ablation studies, four experiments have been performed viz. occluded versus non-occluded faces, gender recognition, emotion recognition, and occlusion recognition. The main motive of this chapter is to present the 3D landmarks-based face restoration technique based on mirroring concept-based epipolar geometry.

Chapter 6: Conclusions and Future Scope

In chapter 6, the conclusion of this thesis is presented. It highlights the contributions made towards the research objectives. The future scope of this research work is discussed.

2. Literature Review

The existing 3D face recognition and reconstruction techniques are studied, along with the research gaps. Based on the research gaps, the objectives and thesis contributions have been presented in this Chapter.

2.1 3D Face Reconstruction Techniques

Figure 2.1 shows different types of 3D face reconstruction techniques. They are mainly of five types viz. 3D morphable model (3DMM), deep learning (DL), epipolar geometry (EG), one shot learning (OSL), and shape from shading. While doing the survey, it was observed that many reconstruction techniques were a hybrid of multiple types out of the techniques mentioned above. Hence, the sixth type is hybrid learning-based reconstruction.

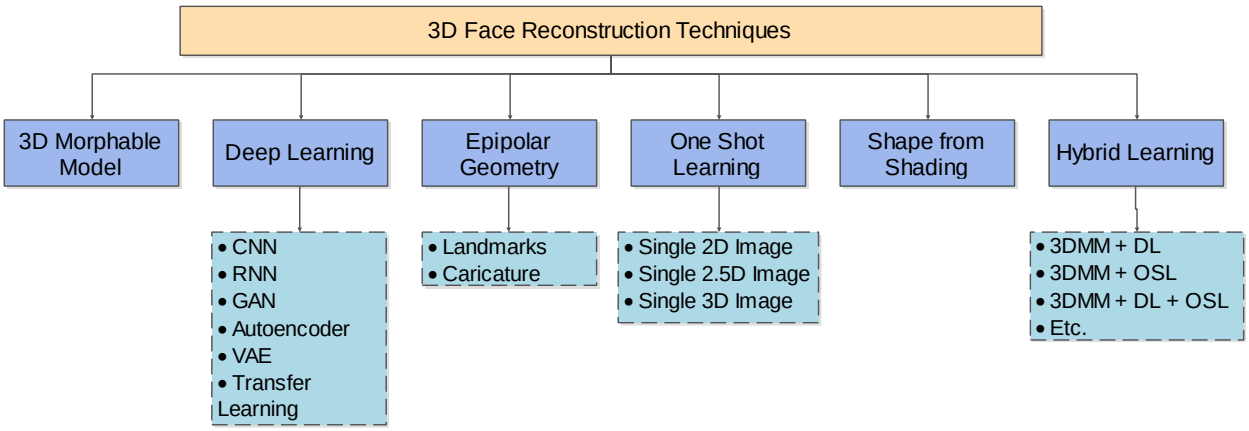


Figure 2.1 Different types of 3D Face Reconstruction Techniques

2.1.1 3D Morphable Model-Based Reconstruction

Maninchedda et al. [99] proposed an automatic reconstruction of a human face and 3D epipolar geometry for eye-glass-based occlusion. A variational segmentation model was proposed, which is capable of representing a wide variety of glasses. Eyeglass detection and segmentation was done based on depth, connectivity, location, and symmetry.

Zhang et al. [100] proposed a technique of reconstructing a dense 3D face point cloud from a single frame of data captured from RGB-D sensor. In the first stage, the initial point cloud of the face region is captured using the K-Mean Clustering algorithm using the aligned depth information and

RGB data information. In the second stage, the artificial neural network (ANN) based calculation estimates the point cloud neighborship. Furthermore, applying the RBF interpolation to get the final estimate of point cloud-based 3D face.

Jiang et al. [101] proposed a 3D face restoration algorithm (PIFR) based on three-dimensional morphable model (3DMM). The input image is normalized to get more information about the face along with the visibility of facial landmarks.

Wu et al. [102] presented a 3D face expression reconstruction technique using a single image. A cascaded regression framework was used to calculate the parameters for 3D morphable models. After the initialization phase, a histogram of oriented gradients (HOG) features and landmark displacement were used for the feature extraction phase.

Kollias et al. [103] proposed a novel technique for synthesizing facial expressions, the degree of positive and negative emotion, and arousal. Based on the valence-arousal (VA) technique, 600K frames were annotated from the 4DFAB dataset [104]. 4DFAB dataset [104] contains 180 subjects aged between 5-75 years.

Lyu et al. [105] proposed a Pixel-Face dataset consisting of high-resolution images using 2D images. 3D morphable model-based Pixel-3DM was proposed for 3D facial reconstruction.

2.1.2 Epipolar Geometry Based Reconstruction

Anbarjafari et al. [106] proposed a novel technique for generating 3D faces captured by phone cameras. A total of 68 facial landmarks were used to divide the face into four regions. Different phases were used during texture creation, weighted region creation, model morphing, and composing.

2.1.3 One-Shot Learning Based Reconstruction

Xing et al. [107] presented 3D face reconstruction technique using a single image without the ground-truth 3D shape. The face model rendering using new viewpoints and the fine-tuning in the supervised mode was used in the process. After 3D face reconstruction, the rigid body transformation, as well as rendering, were performed. To improve the quality of rendering, the bootstrap fine-tuning method was used to send the feedback to 3D face reconstruction face.

2.1.4 Shape from Shading Based Reconstruction

Jiang et al. [19] was inspired by face animation using RGB-D and the monocular video. Initially, the computation of coarse estimation was done for the target 3D face by parametric model fitting to the input image.

2.1.5 Deep Learning Based Reconstruction

Kim et al. [9] proposed a deep convolutional neural network-based 3D face recognition algorithm. 3D face augmentation technique synthesized a variety of facial expressions using a single scan of 3D face.

Gilani et al. [108] proposed a technique for developing a huge corpus for labeled 3D faces of people. They trained a convolutional neural network, namely, FR3DNet to recognize the 3D faces of over 3.1 Million faces of 100K people. The testing was done on 31,860 images of 1853 people.

Thies et al. [95] presented a neural voice puppetry technique for generating photo-realistic output video the source input audio. This was based on DeepSpeech recurrent neural networks using the latent 3D model space. Audio2ExpressionNet was responsible for converting the input audio to a particular facial expression.

Li et al. [109] proposed SymmFCNet, a symmetry consistent convolutional neural network used for missing pixels on the half face for the development of the other half. SymmFCNet consists of two parts, namely, the illumination-reweighted warping subnet and the generative reconstruction subnet.

Han et al. [110] proposed a sketching system that creates the 3D caricature photos by modifying the facial features. An unconventional deep learning method was designed to get the vertex-wise exaggeration map based on the 2D sketch images given by the user. FaceWarehouse dataset [63] was used while training 150 neutral face models, and ten different exaggerations are applied to each model.

Moschoglou et al. [111] implemented an autoencoder like 3DFaceGAN for modeling 3D facial surface distribution. Reconstruction loss and adversarial loss were used during the training of the generator and discriminator.

2.1.6 Hybrid Learning Based Reconstruction

Richardson et al. [22] proposed a technique for generation of the database, which has photo-realistic face images using geometries. Convolutional neural network-based ResNet model [112] was used for building the proposed network. Adam optimizer was used with a fixed learning rate. This technique is unable to restore the images, which have different facial attributes than the training data. Therefore, it is unable to generalize the training process for new face generations.

Liu et al. [113] proposed a 3D face reconstruction technique.

Richardson et al. [75] proposed a single shot learning-based convolutional neural network (CNN) model, extracting the facial shape in a coarse-to-fine technique. There are two parts of the proposed architecture, namely CoarseNet and FineNet. CoarseNet and FineNet were responsible for recovering coarse facial features and refining the facial geometry, respectively.

Jackson et al. [70] proposed a CNN-based model capable of reconstructing 3D face geometry by using a single 2D facial image. This method did not require any kind of facial alignment. It works on various types of expressions and poses. 3D facial geometry was reconstructable, including the missing information without the use of a 3DMM. The convergence function used is Normalized Mean Error (NME) using the cube root of the voxel count.

Tewari et al. [114] proposed a generative model-based convolutional autoencoder network. The encoding part of the network was based on AlexNet [85] and VGGFace [115] model final activation passed to the semantic code vector, and the decoder reconstructs the face.

Dou et al. [23] proposed a deep neural network (DNN) based technique for end-to-end 3D face reconstruction using a single 2D image. Two components of DNN, namely, multi-task loss function and fusion CNN were combined.

Han et al. [116] proposed a sketching system for 3D faces and caricature modeling using CNN-based deep learning. Generally, the rich facial expressions were generated through MAYA and ZBrush. However, it includes the gesture-based interaction with the user. The shape level input was appended with the output of the fully connected layer to generate the bilinear output.

Hsu et al. [117] proposed two different approaches for cross-pose face recognition. One of the techniques is based on 3D reconstruction, and the other is built using deep CNN. The different components of the face are built out of the gallery of 2D faces. 3D surface is reconstructed using 2D face components. The recognition is done using the components of the testing face image.

Feng et al. [20] developed a technique (FaceLFnet) to restore 3D face using Epipolar Plane Images

(EPI) with the use of CNN for recovery of vertical and horizontal 3D face curves. The photorealistic light field images were synthesized using the 3D faces. A total of 14K facial scans of 80 different people were used during the training process, making up to 11 Million facial curves/EPIs.

Zhang et al. [118] proposed a technique for 3D face reconstruction using the combination of morphable faces and sparse photometric stereo. An optimization technique was used for per-pixel lighting direction along with the illumination at high precision. It was followed by semantic segmentation on input images and geometry proxy to reconstruct details such as wrinkles, eyebrows, whelks, pores comparable to movie quality productions. Synthetic as well as real data was used in the implementation. The average geometric error was used for verifying the quality of reconstruction.

Tran et al. [119] proposed a bump map-based 3D face reconstruction technique which can handle extreme occlusions. The convolutional encoder-decoder method was used while estimating the bump maps. Max-pooling and rectified linear unit (ReLU) were used along with the convolutional layers. ReLU adds the non-linearity into model training, therefore reducing the underfitting. Max-pooling is used as a function to reduce the trainable parameters.

Feng et al. [120] presented the benchmark dataset consisting of 2K faces in 2D for 135 people and their corresponding 3D face scans as the ground truth. Five different approaches for 3D face reconstruction were evaluated for the competition.

Feng et al. [121] proposed a 3D face reconstruction technique called Position map Regression Network (PRN) based on the UV position maps generated by 2D representations. A convolutional neural network regressed the 3D shape from a one-shot 2D image. The weighted loss function used different weights in the form of a weight mask while doing the convolutions.

Liu et al. [122] proposed an encoder-decoder-based network used for regressing 3D face shape from 2D images. The joint loss calculation is done based on 3D face reconstruction as well as identification errors.

Chinaev et al. [123] developed CNN based model for 3D face reconstruction in real-time using mobile devices. MobileFace CNN was used for the testing phase.

Gecer et al. [39] proposed a 3D face reconstruction technique based on deep convolutional neural networks (DCNNs) and generative adversarial networks (GANs). In UV space, GAN was used to train the generator for facial texture. An unconventional 3DMM fitting strategy was formulated

based on differentiable renderer and GAN.

Deng et al. [124] presented a CNN-based single-shot face reconstruction method for weakly supervised learning. The perception level loss and image-level loss were combined during the experimentation. Shape confidence learning for the multi-shot face reconstruction technique was studied.

Yuan et al. [125] proposed a 3D face restoration technique for occluded faces using the inverse of 3DMM and GAN. Local discriminator and global discriminator were used for verifying the quality of the generated 3D face.

Luo et al. [126] implemented a Siamese CNN method for 3D face restoration. The training of the method utilized the triplet loss principle-based contrastive loss. Two loss functions, namely, weighted parameter distance cost (WPDC) and contrastive cost function, were used to validate the quality of the reconstruction method.

Gecer et al. [127] proposed a GAN-based method to synthesize high-quality 3D faces based on the coupled UV maps. The UV map of texture, shape, and normals was used in the synthetic 3D face and input to the discriminator for the classification of real and fake images. Conditional GAN was used for expression augmentation. For training data, 10K new individual identities were randomly synthesized from 300W-LP dataset.

Chen et al. [128] proposed a 3D face reconstruction technique in a self-supervised 3DMM trainable VGG encoder using a single image. A two-stage framework was implemented to regress the 3DMM parameters for reconstructing facial details. The CelebA [129] dataset is used for the training, and LFW [130] dataset is used along with the CelebA while testing.

Ren et al. [131] developed an encoder-decoder framework for video deblurring of 3D face points. The identity knowledge and facial structure were predicted by the rendering branch and 3D face reconstruction. In the case of differentiable rendering, ResNet-50 was employed.

Tu et al. [132] developed a 2D-assisted self-supervised learning (2DASL) technique that can used for the 2D face images having noisy information of landmarks to improve the quality of 3D face models. A novel method based on self-critic learning was developed to improve the 3D face model learning procedure. The two datasets, namely AFLW-LFPA [133] and AFLW2000-3D [134] have been compared for 3D face restoration and face alignment.

Liu et al. [135] proposed an automatic method for generating pose-and-expression-normalized (PEN) 3D faces. The overall facial recognition had improved invariant of expression and posed

using the PEN 3D faces. The proposed technique gave state-of-the-art results for the reconstruction as well as the facial alignment.

Lin et al. [67] implemented a 3D face reconstruction technique based on the one-shot image in the wild. Graph convolutional networks were used for generating the high-density facial texture. FaceWarehouse [63] along with the CelebA [129] database were used for training purposes.

Ye et al. [136] presented a big dataset of 3D caricatures and generated a PCA-based linear 3D morphable model for the caricature shapes. 6.1K portrait caricature images were collected from pinterest.com as well as the WebCaricature dataset [137].

Lattas et al. [138] proposed a technique for producing high-quality 3D face reconstruction using arbitrary images. The large-scale database was collected using 200 different subjects based on their geometry and reflectance. The image translation networks were trained to estimate specular and diffuse albedo along with the specular normals.

Zhang et al. [139] proposed an automatic landmark detection and 3D face restoration for the caricatures. The input 2D image of the caricature was used for regressing the orientation and shape of 3D caricature. The ResNet model was used for encoding the input image to a latent space. The decoder was used along with the fully connected layer to generate the projection of 3D landmarks on the caricature. Different loss functions have been used for landmarks, caricature shape, and projection parameters.

Deng et al. [140] presented DISentangled precisely-COntrollable (DiscoFaceGAN) latent embedding for representing fake people with various poses, expressions, and illumination. Contrastive learning was employed for promoting disentanglement by doing the comparison of rendered faces with the real ones.

Li et al. [141] proposed a 3D face reconstruction technique, which used 2D and 3D landmarks to estimate the pose of 3D face using coarse-to-fine estimation. Further, it used an adaptive reweighting method to generate 3D model.

Chaudhuri et al. [142] proposed a deep learning method to train personalized dynamic albedo maps and the expression blend shapes. Hence, the 3D face restoration was generated in a photorealistic manner. The two constraints for the model training proposed were face parsing loss and reflectance and blendshape gradient loss was responsible for capturing the semantic meaning of the reconstructed blend shapes.

Shang et al. [143] proposed a self-supervised learning technique for occlusion aware view

synthesis. Three different loss functions namely, depth consistency loss, pixel consistency loss, and landmark-based epipolar loss, were used for multi-dimensional consistency.

Cai et al. [144] proposed an Attention Guided GAN (AGGAN), which is capable of 3D facial reconstruction using a single depth (2.5D) image. AGGAN generates 3D voxel image from the depth image using the autoencoder technique.

Xu et al. [145] proposed a technique for training the head geometry model without using 3D ground-truth data. The combination of deep synthetic image with head geometry was trained using CNN without optimization. The head pose manipulation was done using GANs and 3D warping.

Table 2.1 presents the detailed comparative analysis of 3D facial reconstruction techniques, including the approaches and models. Table also shows alignment done in the method, along with the binary value for the presence of convergence, deep learning, and synthetic data.

Table 2.1: Comparative analysis of 3D facial reconstruction techniques

Reference	Year	Approaches/Models used	Is Face Alignment Done?	Convergence factor	Is Deep Learning Done?	Synthetic Data used?
[22]	2016	SFS with landmarks and deep learning	No	MSE	Yes	Yes
[113]	2017	3DMM and SFS, learning cascaded regression	Yes	Mean Absolute Error (MAE)	No	Yes
[75]	2017	CNN and Coarse-to-fine details	No	MSE	Yes	Yes
[70]	2017	Volumetric regression networks (Multitask and Guided)	Yes	Normalized Mean Error (NME)	Yes	No
[114]	2017	Auto encoder-based CNN	Yes	Geometric, Photometric, and Landmark error	Yes	No
[23]	2017	DNN architecture	No	Root Mean Square Error (RMSE)	Yes	No
[116]	2017	CNN based deep regression network	No	Mean error	Yes	Yes
[99]	2017	Automatic reconstruction of a human face and 3D epipolar geometry	Yes	Mean and Standard Deviation	No	No
[9]	2018	3D deep feature vector and 3D augmentation of faces	Yes	Cumulative Matching Characteristic (CMC) and Receiver Operator Characteristic (ROC) curve	Yes	Yes
[19]	2018	Coarse face modeling, Medium face modeling, and Fine face modeling	Yes	RMSE	No	No
[100]	2018	Clustering and interpolation-based reconstruction	No	Error distribution	No	No
[117]	2018	Faster Region-based CNN (RCNN) and reduced tree structure model	No	Moving Least Squares (MLS)	Yes	Yes
[108]	2018	FR3DNet, CNN, Data augmentation. Main objective of the paper is to close the gap between size of 2D and 3D datasets for effective 3DFR	Yes	ROC curve	Yes	Yes
[20]	2018	FaceLFNet. 3DMM with facial geometry. EPIs and CNN	No	RMSE	Yes	No

[118]	2018	Leverages sparse photometric stereo (PS)	No	Average geometric error for reconstruction.	No	Yes
[119]	2018	Deep convolution encoder-decoder	No	ROC curve	No	No
[120]	2018	3D dense face reconstruction algorithm. 3DMM-CNN	No	RMSE	Yes	No
[121]	2018	UV position maps	Yes	NME	Yes	No
[122]	2018	Encoder-decoder network	No	RMSE	Yes	No
[101]	2018	PIFR based 3DMM	Yes	Mean euclidean metric (MEM)	No	No
[102]	2019	3DMM. Cascaded regression.	No	RMSE and MAE	No	No
[106]	2019	Blended model	Yes	Structural Similarity Index Metric (SSIM)	No	No
[123]	2019	MobileNet CNN	No	Area Under the Curve (AUC)	Yes	No
[107]	2019	Deep learning model	Yes	NME	Yes	Yes
[39]	2019	3DMM fitting based on GANs and a differentiable renderer	No	Mean and Standard Deviation	Yes	No
[124]	2019	CNNs	Yes	RMSE	Yes	No
[125]	2019	Inverse 3DMM GAN model	Yes	Peak Signal to Noise Ratio (PSNR) and SSIM	Yes	Yes
[126]	2019	Siamese network-based CNN model	Yes	ROC curve	Yes	No
[127]	2019	GAN	No	Wasserstein GAN (W-GAN) adversarial loss	Yes	Yes
[128]	2019	Self-supervised 3DMM encoder	Yes	RMSE	No	No
[131]	2019	Encoder-decoder framework	Yes	PSNR and SSIM	Yes	Yes
[95]	2019	Neural rendering	No	RMSE	Yes	Yes
[103]	2020	Blended model	No	Pearson correlation and MSE	Yes	Yes
[109]	2020	SymmFCNet	Yes	PSNR, SSIM, Identity Distance, and Perceptual Similarity	Yes	No
[110]	2020	Deep learning-based technique	Yes	MSE	Yes	Yes
[132]	2020	2D-Assisted Self Supervised Learning (2DASL)	Yes	NME	Yes	Yes
[111]	2020	VAE-GAN	Yes	Normalized dense vector error	Yes	No
[135]	2020	Summation model and cascaded regression	Yes	MAE and NME	No	No
[67]	2020	Graph convolutional networks	No	W-GAN adversarial loss	Yes	No
[136]	2020	PCA model	No	Adversarial loss, Bidirectional cycle-consistency loss, Cross-domain character loss, and User control loss	Yes	Yes
[138]	2020	Generation of per-pixel diffuse and specular components	No	PSNR	Yes	Yes
[139]	2020	Parametric model based on vertex deformation space	Yes	Cumulative error distribution	Yes	Yes
[140]	2020	DiscoFaceGAN	Yes	Adversarial loss, Imitative loss, and Contrastive loss.	Yes	Yes
[141]	2020	Joint 2D and 3D metaheuristic method	Yes	3D Root Mean Square Error (3DRMSE)	No	No
[142]	2020	End-to-end deep learning framework	Yes	NME and AUC	Yes	No
[143]	2020	MGCNet	Yes	RMSE	No	No
[105]	2020	3D morphable model-based Pixel-3DM	Yes	NME and RMSE	No	Yes
[144]	2020	Attention Guided Generative Adversarial Networks (AGGAN)	Yes	Intersection-over-Union (IoU) and Cross-	Yes	No

				Entropy Loss (CE)		
[145]	2020	GAN	Yes	Adversarial loss	Yes	Yes
[87]	2020	Variational autoencoder, bi-LSTM, and triplet loss training	No	MAE	Yes	No
[146]	2020	Deep learning process, game-theory based generator and discriminator	No	MAE	Yes	No
[147]	2021	Variational autoencoders and triplet loss training	No	MAE	Yes	No
[148]	2021	Single-shot learning-based weakly supervised multi-face reconstruction technique	Yes	NME	Yes	No

The detailed pros and cons of all the reconstruction techniques in the literature have been summarized in Table 2.2.

Table 2.2 Pros and cons of existing 3D face reconstruction techniques

References	Year	Pros	Cons
[22]	2016	Efficient on various illumination conditions and expressions Generalizes well on the synthetic data	Fail to reconstruct new faces in the test set Fails on facial attributes not present in synthetic data
[113]	2017	No annotation needed for invisible landmarks	The facial texture quality is low Unable to apply in real-world scenarios
[75]	2017	Fine details such as wrinkles can be reconstructed	Fails to generalize on facial features not in training data Dependency on synthetic data
[70]	2017	Reconstruction is done using single 2D image	Lack of facial alignment leads to generation of almost identical faces after reconstruction
[114]	2017	End-to-end deep learning	Fail under occlusion such as beard or external object Shrinked reconstruction
[23]	2017	Simplified framework with end-to-end model instead of iterative model	Dependency on synthetic data
[116]	2017	Free hand sketch of face gets converted to 3D caricature model	Unnatural results when exaggeration is inconsistent
[99]	2017	Model training can be done on mobile A general variational segmentation model is proposed to generalize the glasses	Occlusion-invariant only on glasses
[9]	2018	Transfer Learning based model is faster to train	Loses information while converting 3D point cloud to 2.5D
[19]	2018	3D face reconstructed from single 2D image	SFS technique depends on pre-defined knowledge about facial geometry such as facial symmetry
[100]	2018	100K point clouds are generated with single shot of RGB-D sensor Low-cost method	External hardware is required
[117]	2018	3D component-based approach requires only few 3D models CNN based models can easily handle in-the-wild characteristics	3D component-based approach does not generalize well CNN based approach requires large dataset for training
[108]	2018	Deep CNN model trained on 3.1M 3D faces of 100K subjects	Training a CNN from scratch is time consuming
[20]	2018	Method generalizes well This model free approach is superior choice in medical applications	Huge amount of epipolar plane image curves are required
[118]	2018	High quality 3D faces are generated with fine details	Dependent on light falling on face
[119]	2018	Bump map regression takes 0.03 s/image Face segmentation requires 0.02 s/image	Unoptimized soft symmetry implementation takes 50 s/image

[120]	2018	3D face reconstruction from single 2D image	Texture based reconstruction takes high computational time
[121]	2018	Size of the model is 160 MB in contrast to 1.5 GB of Volumetric Regression Network The UV position map can generalize well	Unable to apply in real world scenarios
[122]	2018	It is a light weight network	The joint loss function affects the quality of face shapes
[101]	2018	Good reconstruction invariant of poses	The reconstruction needs improvement for large poses
[102]	2019	The reconstruction technique is robust to light variation	Less landmarks are available, hence landmark displacement features can be improved
[106]	2019	Good generalization through UV map based on feature points	Dependent on database with good head shapes Due to this the overall head shape is lacking good quality
[123]	2019	Fast training on mobile devices Realtime application	Annotation of 3D faces using morphable model is costly at preprocessing stage of proposed MobileNet
[107]	2019	Rendering of input image to multiple view angles 2D image to 3D shape is reconstructed	Rigid-body transformation is used for preprocessing
[39]	2019	High texture 3D images are generated using UV maps with GANs	GANs are hard to train Cannot be applied in real time solutions
[124]	2019	Large pose and occlusion invariant Weakly supervised learning	Confidence of model is low on occlusion during prediction
[125]	2019	Synthetic faces under occlusion images generated with semantic mapping to facial landmarks	Multiple discriminators add on to the model complexity GANs are hard to train
[126]	2019	Siamese CNN based reconstruction has achieved at-par normalized mean error when compared to 3D dense face alignment method	Face recognition has not been tested in-the-wild The number of images in training set are low
[127]	2019	High quality 3D faces are generated with fine details	GANs are hard to train Cannot be applied in real time solutions
[128]	2019	Faces are generated with good quality under normal occlusion Details of face are captured using UV space	Model fails on extreme occlusion, expression, and large pose
[131]	2019	Face deblurring is done over video handling challenge of pose variation	High computational cost
[95]	2019	Technique is fast to train as it depends on transfer learning and not training from scratch	Video quality needs to be improved for real world applications
[103]	2020	Works for in-the-wild face datasets	4DFAB dataset is not publicly available
[109]	2020	Missing pixels are regenerated using generative model	Dependency on multiple networks
[110]	2020	2D image as input can be converted to caricature of 3D face model	Caricature quality is affected when occlusion such as eyeglasses exist Not invariant to lighting conditions
[132]	2020	Works for in-the-wild 2D faces along with the noisy landmarks. Self-supervised learning	Dependency on 2D-to-3D landmarks annotation
[111]	2020	3D GAN method for generation and translation of 3D face High frequency details are preserved	GANs are hard to train. Cannot be applied in real time 3D face solutions
[135]	2020	3D face reconstruction from single 2D image 3D face recognition invariant of pose and expression	Does not include occlusion invariance
[67]	2020	No large-scale dataset is required Detailed and colored 3D mesh image	Does not work for occluded face regions
[136]	2020	End-to-end deep learning method 6.1K 3D meshes of caricature are synthesized Generates high quality caricatures	Does not work well if the input image is occluded
[138]	2020	Generates high resolution avatars using GANs	Fails to generate avatars of dark skin subjects. Minor alignment errors lead to blur of pore details
[139]	2020	3D caricature shape is directly built from 2D caricature image	Does not work well if the input image is occluded
[140]	2020	Face generation is precise over expressions, poses, and illumination	Low quality of model is generated under low lighting and extreme poses

[141]	2020	Robust to partial occlusions and extreme poses	Fails when 2D and 3D landmarks are wrongly estimated for occlusion
[142]	2020	Trained through in-the-wild videos Generates high quality 3D face and facial motion transfer from one person to other	Does not work well under external occlusion
[143]	2020	Reconstruction is done through occlusion-aware method	Does not work well under external occlusion
[105]	2020	Proposed technique can do face analysis and generation effectively	External occlusions are not considered
[144]	2020	2.5D to 3D face mapping is done with attention-based GAN Handles wide range of head poses and expressions	Unable to fully reconstruct the facial expression in case of big open mouth
[145]	2020	CNN-based model learns head-geometries even without ground-truth data	Unable to handle large pose variations
[87]	2020	Mirroring technique is faster for reconstruction as compared to deep learning-based methods The preprocessing time, reconstruction time, and verification time are faster in computation as compared to the state-of-the-art	Preprocessing does not include facial alignment Technique has not been tested on a big dataset of 3D faces
[146]	2020	End-to-end deep learning technique Occlusion invariant reconstruction is done	Not tested for being scalable
[147]	2021	One shot learning based 3D face restoration technique Landmarks based reconstruction is faster as compared to mesh-based reconstruction using the mirroring technique	Facial alignment is missing Maximum size of the dataset is 4666 3D images. Deep learning model can be trained better if the dataset is huge
[148]	2021	Single network model for multi-face reconstruction Faster preprocessing for feature extraction	Low facial texture Multiple GPUs required

2.2 Performance Evaluation Measures

Evaluation measures are important to know the quality of the trained model. There are various evaluation metrics, namely, mean absolute error (MAE), mean squared error (MSE), normalized mean error (NME), root mean squared error (RMSE), cross-entropy loss (CE), area under the curve (AUC), intersection over union (IoU), peak signal to noise ratio (PSNR), receiver operator characteristic (ROC), and structural similarity index (SSIM). Table 2.3 presents the evaluation of 3D face reconstruction techniques in terms of performance measures. The most important performance measures are MAE, MSE, NME, RMSE, and adversarial loss during facial reconstruction. These are the five widely used performance measures. Adversarial loss is being used since 2019 with the advent of GANs in 3D images.

Table 2.3 Summary of performance evaluation measures in literature

References	Year	Adversarial Loss	AUC	CE	IoU	MAE	MEM	MSE	NME	PSNR	RMSE	ROC	SSIM	Other
[22]	2016	×	×	×	×	×	×	✓	×	×	×	×	×	×
[113]	2017	×	×	×	×	✓	×	×	×	×	×	×	×	×
[75]	2017	×	×	×	×	×	×	✓	×	×	×	×	×	×
[70]	2017	×	×	×	×	×	×	×	✓	×	×	×	×	×

[114]	2017	×	×	×	×	×	×	×	×	×	×	×	×	✓
[23]	2017	×	×	×	×	×	×	×	×	×	✓	×	×	×
[116]	2017	×	×	×	×	×	×	×	×	×	×	×	×	✓
[99]	2017	×	×	×	×	×	×	×	×	×	×	×	×	✓
[9]	2018	×	×	×	×	×	×	×	×	×	×	✓	×	✓
[19]	2018	×	×	×	×	×	×	×	×	×	✓	×	×	×
[100]	2018	×	×	×	×	×	×	×	×	×	×	×	×	✓
[117]	2018	×	×	×	×	×	×	×	×	×	×	×	×	✓
[108]	2018	×	×	×	×	×	×	×	×	×	×	✓	×	×
[20]	2018	×	×	×	×	×	×	×	×	×	✓	×	×	×
[118]	2018	×	×	×	×	×	×	×	×	×	×	×	×	✓
[119]	2018	×	×	×	×	×	×	×	×	×	×	✓	×	×
[120]	2018	×	×	×	×	×	×	×	×	×	✓	×	×	×
[121]	2018	×	×	×	×	×	×	×	✓	×	×	×	×	×
[122]	2018	×	×	×	×	×	×	×	×	×	✓	×	×	×
[101]	2018	×	×	×	×	×	✓	×	×	×	×	×	×	×
[102]	2019	×	×	×	×	✓	×	×	×	×	✓	×	×	×
[106]	2019	×	×	×	×	×	×	×	×	×	×	×	✓	×
[123]	2019	×	✓	×	×	×	×	×	×	×	×	×	×	×
[107]	2019	×	×	×	×	×	×	×	✓	×	×	×	×	×
[39]	2019	×	×	×	×	×	×	×	×	×	×	×	×	✓
[124]	2019	×	×	×	×	×	×	×	×	×	✓	×	×	×
[125]	2019	×	×	×	×	×	×	×	×	✓	×	×	✓	×
[126]	2019	×	×	×	×	×	×	×	×	×	×	✓	×	×
[127]	2019	✓	×	×	×	×	×	×	×	×	×	×	×	×
[128]	2019	×	×	×	×	×	×	×	×	×	✓	×	×	×
[131]	2019	×	×	×	×	×	×	×	×	✓	×	×	✓	×
[95]	2019	×	×	×	×	✓	×	×	×	×	×	×	×	✓
[103]	2020	×	×	×	×	×	×	✓	×	×	×	×	×	×
[109]	2020	×	×	×	×	×	×	×	×	✓	×	×	✓	×
[110]	2020	×	×	×	×	×	×	✓	×	×	×	×	×	×
[132]	2020	×	×	×	×	×	×	×	✓	×	×	×	×	×
[111]	2020	×	✓	×	×	×	×	×	×	×	×	×	×	✓
[135]	2020	×	×	×	×	✓	×	×	✓	×	×	×	×	×
[67]	2020	✓	×	×	×	×	×	×	×	×	×	×	×	×
[136]	2020	✓	×	×	×	×	×	×	×	×	×	×	×	×
[138]	2020	×	×	×	×	×	×	×	×	✓	×	×	×	×
[139]	2020	×	×	×	×	×	×	×	✓	×	×	×	×	×
[140]	2020	✓	×	×	×	×	×	×	×	×	×	×	×	×
[141]	2020	×	×	×	×	×	×	×	×	×	✓	×	×	×
[142]	2020	×	✓	×	×	×	×	×	✓	×	×	×	×	×
[143]	2020	×	×	×	×	×	×	×	×	×	✓	×	×	×
[105]	2020	×	×	×	×	×	×	×	✓	×	✓	×	×	×
[144]	2020	×	×	✓	✓	×	×	×	×	×	×	×	×	×
[145]	2020	✓	×	×	×	×	×	×	×	×	×	×	×	×
[87]	2020	×	×	×	×	✓	×	×	×	×	×	×	×	✓
[146]	2020	✓	×	×	×	✓	×	×	×	×	×	×	×	×
[147]	2021	×	×	×	×	✓	×	×	×	×	×	×	×	✓
[148]	2021	×	×	×	×	×	×	×	✓	×	×	×	×	×

2.3 Dataset used in 3D Face Reconstruction Techniques

Table 2.4 depicts the detailed description of datasets used in 3D face reconstruction techniques. The analysis of different datasets highlights the fact that most 3D face datasets are publicly available datasets. They do not have a high number of images to train the model compared to 2D face publicly datasets. This makes the research in 3D faces more interesting because the scalability factor has not been tested and becomes an active area of research. It is worth mentioning that only

three datasets, namely, Bosphorus, KinectFaceDB, and UMBDB datasets, have occluded images for occlusion removal.

Table 2.4 Detailed review of datasets

Dataset Name	Modalities	Total Images	Total Subjects	Emotion Label Availability	Occlusion Label Availability	Publicly Available	Techniques using the dataset
Annotated faces-in-the-wild (AFW) [149]	2D + Landmarks	468	-	No	No	Yes	[113], [101]
Annotated facial landmarks in the wild (AFLW) [150]	2D + Landmarks	25993	-	No	No	Yes	[101]
AFLW2000-3D [134]	2D + Landmarks	2000	-	No	No	Yes	[121], [107], [125], [132], [135], [141], [143]
AFLW-LFPA [133]	2D + Landmarks	1299	-	No	No	Yes	[132]
Age Database (AgeDB) [151]	2D + Age	16488	568	No	No	Yes	[127]
Basel Face Model (BFM2009) [152]	3D	200	-	No	No	Yes	[113], [19]
Bosphorus [153]	2D, 3D	4666	105	Yes	Yes	Yes	[22], [75], [9], [108], [102]
Binghamton University 3D Facial Expression (BU3DFE) [154]	3D	2500	100	Yes	No	Yes	[113],[108], [20], [122], [135]
Binghamton University 4D Facial Expression (BU4DFE) [154]	3D + Time	606	101	Yes	No	Yes	[108],[20], [123]
Chinese Academy of Sciences Institute of Automation (CASIA-3D) [155]	3D	4624	123	Yes	No	Yes	[9], [108]
CASIA-WebFace [156]	2D	494414	10575	No	No	Yes	[109]
CelebFaces Attributes Dataset (CelebA) [129]	2D	202599	10177	Yes	Yes	Yes	[67], [114],[124], [125], [128], [145]
Celebrities in Frontal-Profile in the Wild (CFP) [157]	2D	7000	500	No	No	Yes	[127]
FaceScape [158]	3D	18760	938	Yes	No	Yes	-
Facewarehouse [159]	3D	-	150	Yes	No	Yes	[114], [19], [124], [110], [67]
Face Recognition Grand Challenge (FRGC-v2.0) [160]	3D	4950	466	Yes	No	Yes	[22],[23], [108],[135], [143]
GavabDB [161]	3D	549	61	Yes	No	Yes	[108]
Helen Facial Feature Dataset (Helen) [162]	2D + Landmarks	2330	-	No	No	Yes	[144]
Hi-Lo [111]	3D	6000	-	No	No	Yes	[111]
IARPA Janus Benchmark A (IJB-A) [163]	2D + Landmarks	5712	500	No	No	Yes	[124]
KinectFaceDB [164]	2D, 2.5D, 3D	936	52	Yes	Yes	Yes	[87], [146]
Labeled Faces in the Wild (LFW) [130]	2D + Landmarks	13233	5749	No	No	Yes	[114], [122], [124], [128], [145]
Labeled Face Parts in the Wild	2D +	3000	-	No	No	Yes	[101]

(LFPW) [165]	Landmarks						
Large Scale 3D Faces in the Wild (LS3D-W) [166]	2D + Landmarks	230000	-	No	No	Yes	[107], [124]
Media Integration and Communication Center Florence (MICC-Florence)[167]	2D, 3D	53	53	No	No	Yes	[122],[107], [124], [141], [143], [121]
Notre Dame (ND-2006) [168]	3D	13450	888	Yes	No	Yes	[108]
Texas 3D Face Recognition Database (TexasFRD) [169]	2D + Landmarks, 2.5D	1149	105	No	No	Yes	[108]
University of Houston Database (UHDB31) [170]	3D	25872	77	No	No	Yes	[23]
University of Milano Bicocca 3D Face Database (UMBDB) [171]	2D, 3D	1473	143	Yes	Yes	Yes	[108]
Visual Geometric Group Face Dataset (VGG-Face) [172]	2D + Time	2.6M	2622	No	No	Yes	[114]
VGGFace2 [115]	2D	3.31M	9131	No	No	Yes	[109]
VidTIMIT [173]	Video + Audio	-	43	No	No	Yes	[131]
VoxCeleb2 [174]	2D + Audio	1.12M	6112	No	No	Yes	[142]
WebCaricature [137]	2D	6042	252	No	No	Yes	[136]
YouTube Faces Database (YTF) [175]	2D + Time	3425	1595	No	No	Yes	[122]
300 Videos in the Wild (300 VW) [176]	2D+Time	218595	300	No	No	Yes	[114], [131]
300 Faces in-the-wild Challenge (300W-3D) [177]	2D + Landmarks	600	300	No	No	Yes	[123], [107], [125]
300 Faces in-the-wild with Large Poses (300W-LP) [134]	2D, 3D	61225	-	No	No	Yes	[127], [124], [145], [144], [70], [126]
3D Twins Expression Challenge (3D-TEC) [178]	3D	428	214	Yes	No	Yes	[9]
4D Facial Behaviour Analysis for Security (4DFAB) [104]	3D+Time	1.8M	180	Yes	No	No	[103],[111]

2.4 Tools and Techniques in 3D Face Reconstruction Techniques

Table 2.5 presents the techniques along with hardware used in terms of the graphical processing unit (GPU), size of random-access memory (RAM), central processing unit (CPU), and brief applications. The comparison highlights the importance of deep learning in 3D face reconstruction. GPUs play a vital role in the deep learning-based models. With the advent of Google Collaboratory, GPUs are freely accessible.

Table 2.5 Comparative analysis based upon technique, hardware, and applications

References	Year	Technique	Hardware	RAM (GB)	Applications
[22]	2016	CNN on synthetic data	GPU	8	Real Images
[113]	2017	Cascaded regression	Intel Core i7	8	Real Images
[75]	2017	CNN on synthetic data	Intel Core i7	8	Facial Reenactment
[70]	2017	Direct volumetric CNN regression	GPU	8	Real Images
[114]	2017	Unsupervised deep convolutional	GPU	8	Real Images

		autoencoder			
[23]	2017	End-to-end Deep Neural Network	GPU	8	Real Images
[116]	2017	Deep learning-based sketching	GPU	8	Avatar, Cartoon characters
[99]	2017	Glass-based explicit modelling	Mobile Phone	4	Real Images
[9]	2018	Deep CNN and 3D augmentation	GPU	8	Captured 3D face reconstruction
[19]	2018	Coarse to fine optimization strategy	Intel Core i7	8	Real Images
[100]	2018	RBF interpolation	Intel Core i7	8	3D films and virtual reality
[117]	2018	Landmark localization and deep CNN	Intel Core i7	8	Real Images
[108]	2018	Deep CNN	GPU	8	Real Images
[20]	2018	Epipolar plane images and CNN	GPU	8	Real Images
[118]	2018	Sparse photometric stereo	Intel Core i7	8	Semantic labeling of face into fine region
[119]	2018	Bump map estimation with deep convolution autoencoder	GPU	12	Real Images
[120]	2018	Morphable model, Basel face model, and Cascaded regression	Intel Core i7	8	Real Images
[121]	2018	Position map regression network	GPU	8	Real Images
[122]	2018	Encoder decoder based	GPU	32	Real Images
[101]	2018	3DMM	Intel Core i7	8	Real Images
[102]	2019	Cascaded regression	Intel Core i7	8	Estimation of high-quality 3D face shape from single 2D image
[106]	2019	Best fit blending	Intel Core i7	16	Virtual reality
[123]	2019	CNN regression	GPU	8	Real time application
[107]	2019	Unguided volumetric regression network	Intel Core i7	8	Real Images
[39]	2019	GANs and Deep CNNs	GPU	11	Image augmentation
[124]	2019	Weakly supervised CNN	GPU	8	Real Images
[125]	2019	GANs	GPU	11	De-occluded face generation
[126]	2019	Siamese CNN	Intel Core i7	8	Real Images
[127]	2019	GANs	GPU	4	Image augmentation
[128]	2019	3DMM	Intel Core i7	8	Easy combination of multiple face views
[131]	2019	Encoder decoder based	GPU	8	Video quality enhancement
[95]	2019	RNN and autoencoder	GPU	11	Video avatars, facial reenactment, video dubbing
[103]	2020	Deep neural networks	Intel Core i7	8	Facial affect synthesis of basic expressions
[109]	2020	Symmetry consistent CNN	GPU	11	Natural Images
[110]	2020	Deep CNN	GPU	11	Expression modeling on caricature
[132]	2020	2D assisted self-supervised learning	Intel Core i7	8	Real Images
[111]	2020	GANs	GPU	32	3D face augmentation
[135]	2020	Cascaded coupled regression	GPU	8	Real Images
[67]	2020	Graph convolutional networks	GPU	11	Generate high fidelity 3D face texture
[136]	2020	GANs	GPU	11	Animation, 3D printing, virtual reality
[138]	2020	3DMM and GAN algorithm	Intel Core i7	16	Generation of 4Kx6K 3D face from single 2D face image
[139]	2020	ANN	GPU	16	Expression modeling on caricature
[140]	2020	GANs and VAEs	GPU	8	Vision and graphics
[141]	2020	Adaptive reweighing based optimization	Intel Core i7	8	Real Images
[142]	2020	3DMM and blendshapes	GPU	8	Personalized reconstruction
[143]	2020	Multiview geometry consistency	Intel Core i7	8	Real Images
[105]	2020	3DMM	Intel Core i7	8	Expression modeling
[144]	2020	Attention guided GAN	Intel Core i7	8	2.5D to 3D face generation
[145]	2020	GANs	GPU	12	Face animation and reenactment
[87]	2020	Clustering, VAE, BiLSTM, SVM	GPU	32	Real Images

[146]	2020	End-to-end deep learning	GPU	32	Real Images
[147]	2021	VAE and Triplet Loss	GPU	32	Real Images
[148]	2021	Encoder decoder	GPU	32	Multiface reconstruction

2.5 Research Gaps

Based on the investigation of literature, the following research gaps were identified.

a) Fast and accurate method for pre-processing of 3D face image

More than 20 percent of face recognition systems cannot detect the face accurately due to noise and variations of illumination. Hence, a fast and accurate method for pre-processing 3D face image is required to achieve better accuracy in the recognition system.

b) Automatic registration technique for 3D face image

The accurate registration of human faces plays a vital role in face recognition. Most automatic face registration algorithms such as Iterative closet point algorithm (ICP) suffer from parameter initialization problems. Hence, an automated 3D face registration without initialization is quite required.

c) Automatic restoration technique for 3D face image

The mask and thresholding concepts are used to restore the occluded part of the face. However, they are not efficient to restore the occluded part. The focus of the thesis is to identify the efficient approach that help in determining the occlusion part of the face and restore the missing part using the available features or information about the face.

d) Restriction on 3D face database

Most of studies have been conducted on datasets in controlled environments with controlled expression. There is almost no work on occlusions caused by hair and surface irregularities caused by facial hair. Hence, there is a need to design 3D face database that includes the face images having occlusion variation.

e) Metaheuristic approaches for 3D face recognition

It has been observed from the literature that only a few optimization techniques have been used for 3D face recognition – face modelling, model estimation, and angle rotation. There is a vast scope of applying these metaheuristic techniques in 3D face feature extraction and restoration.

2.6 Objectives

To address the aforementioned research gaps, the following objectives were identified for the

research work:

1. To develop an efficient pre-processing and registration technique for 3D face images.
2. To develop an efficient restoration technique using metaheuristics.
3. To design a framework to efficiently recognize the human face from restored face image.
4. To test and validate the developed framework using standard 3D face databases.

2.5 Thesis Contribution

To achieve the objectives mentioned above, the following contributions have been mentioned in this thesis.

- A comprehensive survey of various 3D face reconstruction algorithms is presented. This will help the researchers to understand different face reconstruction algorithms.

- A novel voxel-based 3D occlusion-invariant face recognition framework is proposed using game theory and simulated annealing. Voxelization and locality preserving projections are applied to preprocess the mesh image. The highest overall classification accuracy from voxel-based facial recognition is 86.1%. For occlusion invariant facial recognition, the average generated by the proposed technique is 75.5%.

- A novel framework for 3D face recognition using voxel reconstruction is proposed using deep sequential learning. 3D geometry-based mirroring technique is applied to the voxelized image for regenerating the occluded voxels. The 3D face recognition has been done using a series of deep learning models viz. variational autoencoder, BiLSTM, and triplet loss training embedding. In the performance evaluation phase, all predictions are shown with and without reconstruction. In terms of gender identification, the accuracy is greater than 90% following restoration for all three datasets due to binary classification. In emotional identification, the accuracy of the Bosphorus dataset is 94.57% after restoration compared to 83.95% before restoration, i.e., 10.62% improvement in accuracy.

- A novel 3D deep learning-based face recognition framework is proposed using landmark-based face restoration, variational autoencoder, and triplet loss. This technique uses the principle of reflection and mid-face plane for the restoration of the facial landmarks. The experimental findings indicate that the proposed method obtained an 8-10 % increase over the current techniques. The computation period derived from the proposed method is roughly 50% quicker than the other approaches. Furthermore, in ablation experiments, the efficiency of the proposed model is checked for occluded versus non-occluded faces, gender, emotion, and recognition of occlusion. The experimental findings indicate that the suggested methodology is capable of comprehending the occluded face with gender classification.

3. Voxel-based 3D Occlusion-Invariant Face Recognition Using Game Theory and Simulated Annealing

3.1 Introduction

This chapter presents voxel-based occlusion-invariant 3D face recognition framework (V3DOFR) based on game theory and simulated annealing. In V3DOFR approach, 3D meshes are converted to voxel form of sizes 4^3, 8^3, and 16^3. After that, locality preserving projection-based embeddings are computed for removing the sparseness of voxels and generating consistent linear embedding per mesh with size 64x3, 128x3, and 256x3, respectively. The generator of triplets provides the triplets of sizes 64x3x3, 128x3x3, and 256x3x3. The simulated annealing is used to check the threshold value of adversarial triplet loss generated after ensembling losses of different grid sizes. The proposed framework is compared with four well-known methods using three face datasets, namely, Bosphorus, UMBDB, and KinectFaceDB.

3.2 Proposed 3D Face Recognition Framework

This section discusses the motivation followed by voxel-based 3D occlusion invariant face recognition framework.

3.2.1 Motivation

The proposed framework is motivated by the recent success of generative adversarial networks (GANs). The use of voxels makes it possible to include the finer details of 3D face. According to the best of the author's knowledge, little work has been done in voxels and deep learning for 3D face recognition. The proposed framework utilizes the concepts of voxelization, locality preserving projections, triplet loss, simulated annealing, and game theory. In the traditional approach, 3D mesh images are converted into depth images (2.5D) or Epipolar geometry-based multiple 2D images, which are further used to train the conventional neural networks (CNNs) or autoencoders. In contrast to 2D and 2.5D images, the presented work is implemented using voxels in 3D form. In Sharma and Kumar [87], mirroring technique-based face reconstruction was done after

voxelization along with BiLSTM based sequential deep learning. Figure 3.1 shows the comparison between the traditional approach [23], [179], and the proposed 3D face recognition framework approach. The proposed approach uses voxels in contrast to depth images or epipolar geometry images.

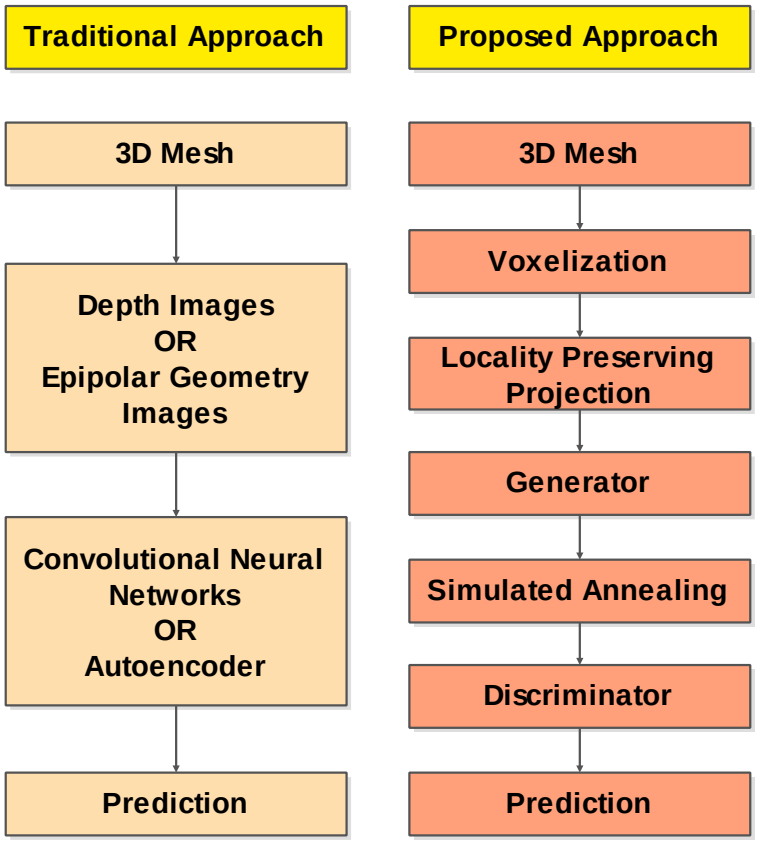


Figure 3.1. Comparison between the traditional approach and proposed approach of 3D face recognition framework

3.2.2 Proposed 3D Face Recognition Framework

The proposed 3D face recognition framework consists of two phases, such as training and testing. Figure 3.2 presents the proposed 3D face recognition framework.

3.2.2.1 Training Phase

There are two sub-phases in the training phase, namely, pre-processing and simulated annealing-based deep learning. The detailed descriptions of these phases are mentioned in the preceding subsections.

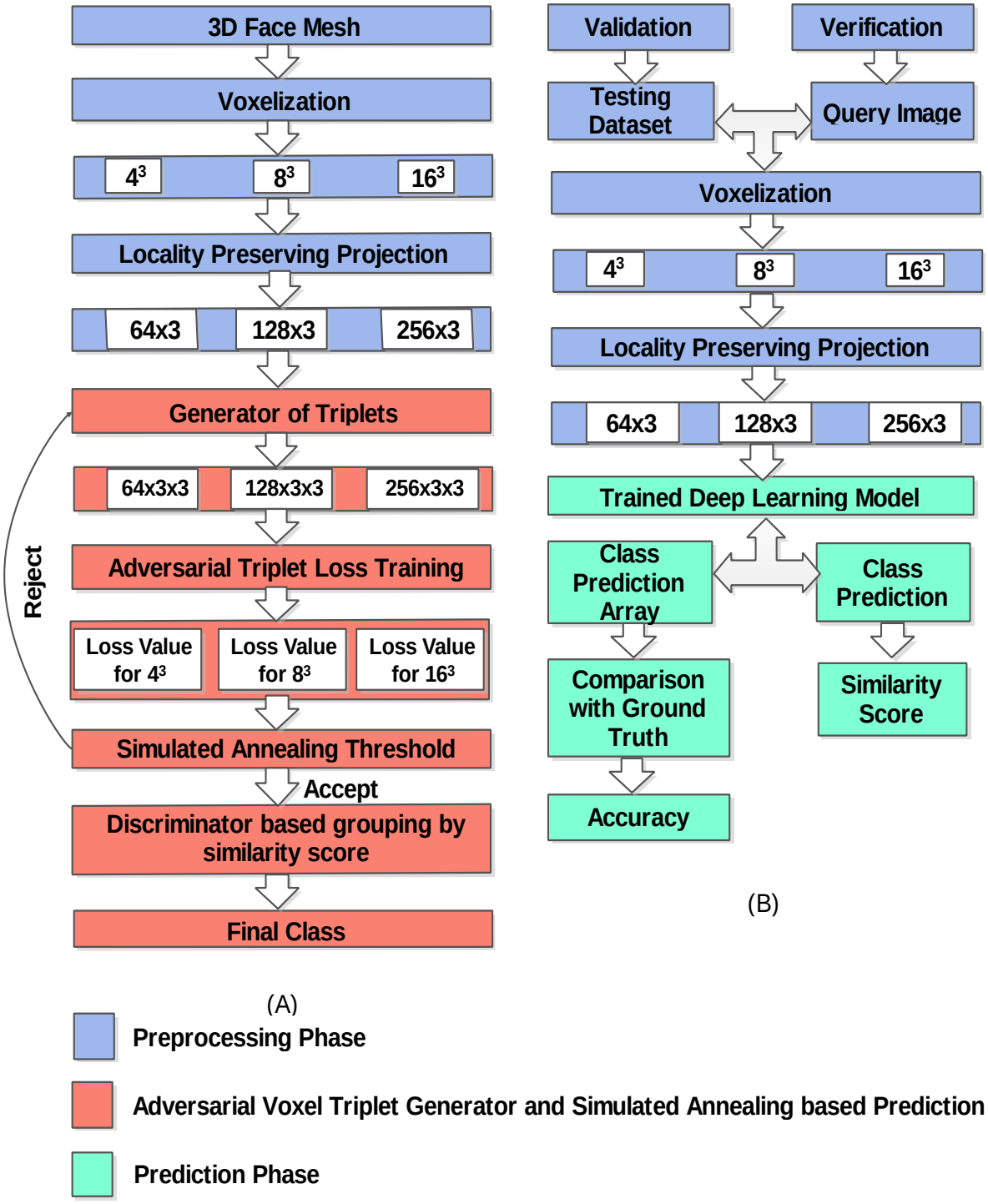


Figure 3.2. Proposed 3D face recognition framework (A) Training phase (B) Testing phase

1. Pre-processing

During the training phase, voxelization and locality preserving projections are two well-known

preprocessing techniques for generating embeddings. Figure 3.3 shows the mesh images and their corresponding voxel images. The voxelization process converts a 3D mesh into voxel form in such a way that 3D coordinates are generated for each triangular mesh represented using cubes in different grid sizes. A single mesh is converted into three different voxel grid sizes viz 4^3, 8^3, and 16^3. The number of voxels generated is sparse for different phases, even for the same size grid. Locality preserving projections are used to handle the problem of sparseness. 4^3 voxels are converted into 64x3 embedding, 8^3 voxels are converted into 128x3 embedding, and 16^3 voxels are converted into 256x3 embedding. Ensembling is a famous technique in making the prediction model more robust towards new test images. Hence, three different kinds of grid sizes are used. It helps in boosting the quality of training data during the preprocessing step.

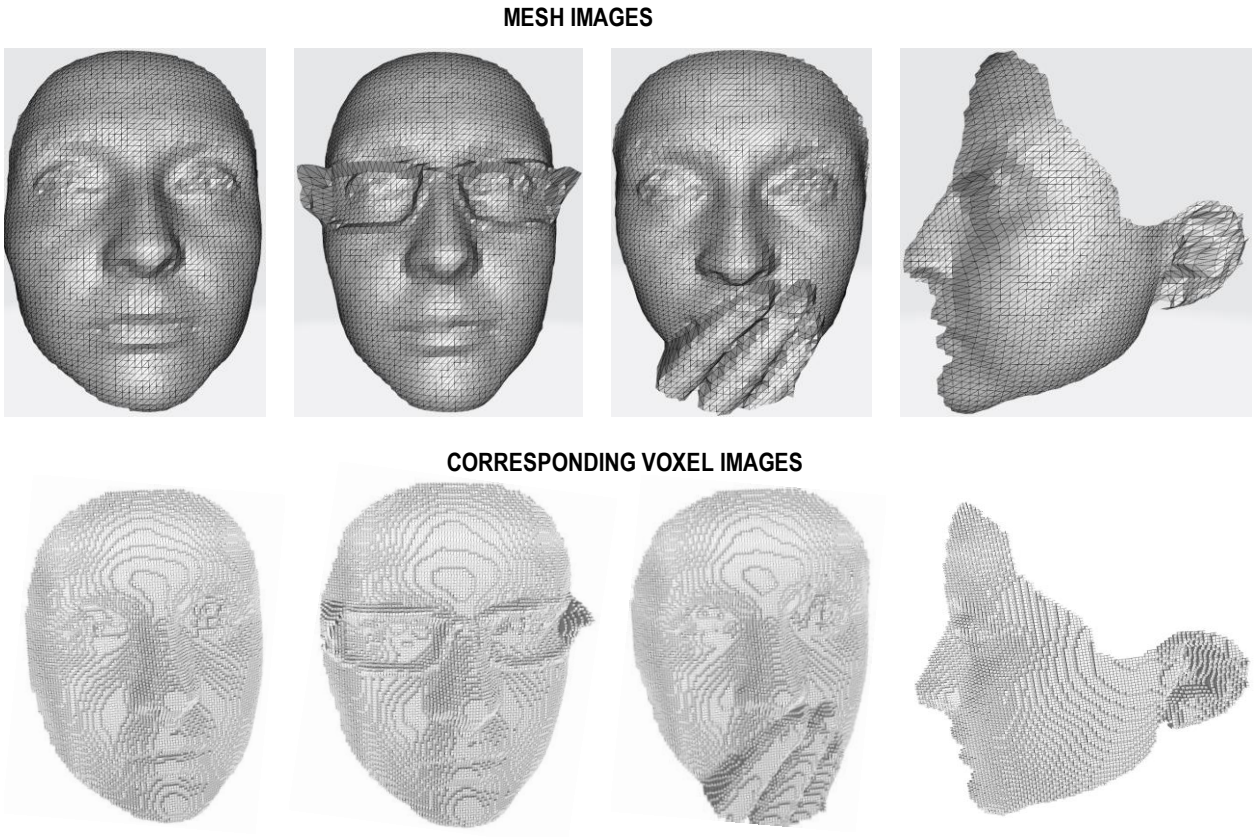


Figure 3.3. Mesh images and its corresponding voxel images

2. Adversarial Voxel Triplet Generator and Simulated Annealing based Prediction

The pre-processing sub-phase produces normalized voxel embedding for further processing. The generator produces triplets of Anchor (A), Positive (P), and Negative (N) for triplet loss training.

Motivated from [180], normalized voxel embeddings for a voxelized mesh image x is represented as $V(x) \in \mathbb{R}^L$. Given $< A, P, N >$ as a triplet, $< A, P >$ is relevant (positive) pair and $< A, N >$ is irrelevant (negative) pair. The objective function to train $V(x)$ such that minimizing the following loss:

$$L_{V,tri} = [d(V(A), V(P)) - d(V(A), V(N)) + m]_+ \tag{3.7}$$

where $d(x, y) = \| \frac{x}{\| x \|} - \frac{y}{\| y \|} \|^2$ is squared Euclidean distance between two L2-normalized vectors.

m is the least margin required between d(A, P) and d(A, N) during training, and $[.]_+ \triangleq \max(., 0)$ denotes the positive component of the input. Let the adversarial voxel triplet generator (G) generates an adversarial sample $G(V(x)) \in \mathbb{R}^L$ by modifying the feature representation $V(x)$ of an image x. While generator training to minimize the triplet loss, G produces hard triplet examples by pushing away the same category vectors and pushing close the different category vectors.

The following objective is to be minimize the adversarial voxel triplet loss during training G,

$$L_{G,tri} = [d(G(V(A)), G(V(N))) - d(G(V(A)), G(V(P))) + m]_+ \tag{3.8}$$

Finally, with a fixed G objective function for training becomes

$$L_{V,tri} = [d(G(V(A)), G(V(P))) - d(G(V(A)), G(V(N))) + m]_+ \tag{3.9}$$

Here, $L_{G,tri}$ and $L_{V,tri}$ makes up an adversarial loss pair. Comparing Eq. (3.7) and Eq. (3.9), V is trained through the triplets generated by G pushing $< A, P >$ closer and $< A, N >$ apart to meet margin m.

The adversarial mechanism using a generator (G) is insufficient without the use of discriminator along with it. The role of discriminator (D) is to monitor and constrain the triplet generator G for attaining a low value of $L_{G,tri}$. Using a discriminator (D), a feature vector is categorized into $(C+1)$ classes, where real class examples are represented by the first C categories and final one denotes the fake class. The triplet $< A, P, N >$ has the labels $< l_A, l_P, l_N >$. The positive pair has $l_A = l_P$ and the negative pair has $l_A \neq l_N$. The loss function is minimized for training the discriminator (D).

$$L_D = L_{D,real} + \beta L_{D,fake} \tag{3.10}$$

Here, D is forced for classification of the triplet's feature vectors correctly by ($L_{D,real}$). $L_{D,real}$ is

mathematically defined as:

$$L_{D,real} = [L_{ll}(D(V(A)), l_A) + L_{ll}(D(V(P)), l_P) + L_{ll}(D(V(N)), l_N)] * 0.33 \quad (3.11)$$

where L_{ll} signifies the log loss. However, the second term $L_{D,\,fake}$ enables D to differentiate between real features and the generated features. The mathematical formulation of $L_{D,\,fake}$ is as follows:

$$L_{D,\,fake} = [L_{ll}(D(G(V(A))), l_{fake}) + L_{ll}(D(G(V(P))), l_{fake}) + L_{ll}(D(G(V(N))), l_{fake})] * 0.33 \quad (3.12)$$

Here, fake class is denoted by l_{fake}. D plays a crucial role in helping G for the preservation of a class of input features. Hence, the subsequent loss enforces the class preservation assumption and represented as:

$$L_{G,class} = [L_{ll}(D(G(V(A))), l_A) + L_{ll}(D(G(V(P))), l_P) + L_{ll}(D(G(V(N))), l_N)] * 0.33 \quad (3.13)$$

The final loss value is minimized by training the voxel triplet generator G and defined as:

$$L_G = L_{G,tri} + \gamma L_{G,class} \quad (3.14)$$

Based on the mean triplet loss for multiple grid sizes, simulated annealing threshold is applied for accepting the predicted similarity score. The concept of simulated annealing was introduced to ensure that the minimization problem of mean loss value as the output from adversarial triplet loss training under discriminator is handled in an effective way by keeping a check on the threshold value. If the mean loss value does not satisfy the simulated annealing threshold, then the embedding is dropped and sent back to the generator for new triplet generation. The similarity score and final class are generated through discriminator for classifying the selected embeddings. Figure 3.4 depicts the prediction and score matching using adversarial triplet loss and simulated annealing. In this figure, M is the number of embeddings after the voxel normalization, and n is the number of triplets forming via generator.

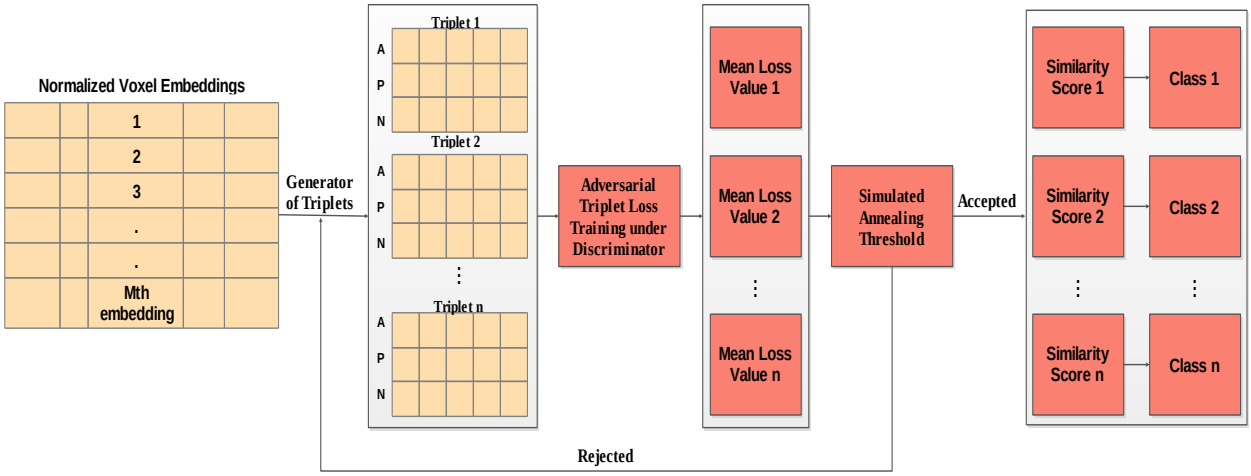


Figure 3.4. Prediction of face using adversarial voxel triplet generator and simulated annealing

3.2.2.2 Testing Phase

There are two sub-phases in the testing phase, namely, pre-processing and the prediction for validation and verification. Figure 3.5 shows the pre-processing and prediction phase for different grid size voxels.

1. Pre-processing

The pre-processing during testing phase is considered for validation/verification. For validation, the testing dataset is considered and voxelization process is carried out on each image in the testing dataset. For verification, the voxelization process is carried out on a single query image. Locality preserving projection normalizes the voxels by removing their sparseness.

2. Prediction

For validation, an array of class predictions is given as output from trained deep learning model. The output array values are compared with ground truth values to compute the accuracy of the proposed model. For verification, the predicted value is a single class and the final similarity score is calculated using the correlation value.

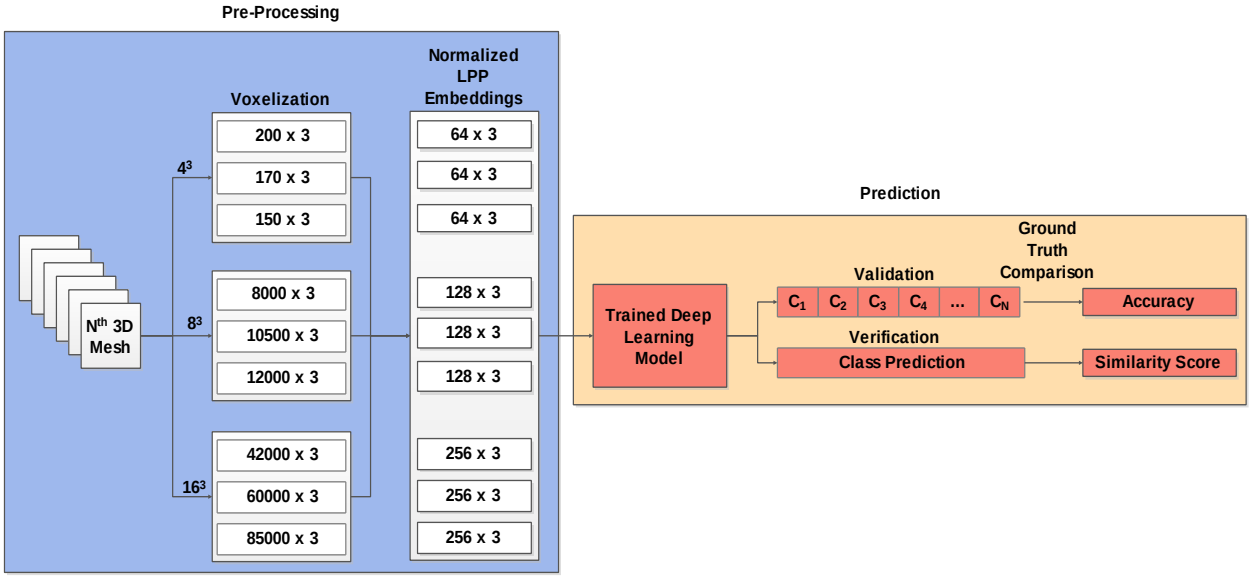


Figure 3.5. Pre-processing and prediction processes in the testing phase

3.2.3 V3DOFR and Computational Complexity

The proposed voxel-based 3D occlusion invariant face recognition (V3DOFR) consists of five steps. Firstly, raw 3D mesh image is taken as input, and the number of triangular units of mesh is counted. If there are no triangular units found, then an error message is generated (see Step 1). In Step 2, voxelization is performed for different grid sizes. The number of voxels and grid sizes are linearly proportional. During the process of voxelization, there is an inconsistency in voxel count due to different poses with in the same class. To overcome this inconsistency, locality preserving projection (LPP) is used in Step 3 that will help in removing the sparseness while maintaining the neighboring voxel properties. Thus, LPP is an effective technique than principal component analysis (PCA) for dimensionality reduction in maintaining the voxel properties at facial feature level. Different grid sizes are converted into different number of LPP feature set. Once the LPP embeddings are generated, triplet generation is followed using generator in Step 4. The generator randomly selects Anchor (A), Positive (P), and Negative (N) embeddings. Deep learning-based triplet loss training is performed for computation of loss value. Further, the normalization of loss values for corresponding grid sizes is performed in Step 4. In final step, the average of normalized loss value is calculated for simulated annealing-based triplet selection. After triplet selection, discriminator assigns the class identification number. If triplet is not selected, then new triplet is generated through generator. Algorithm 3.1 depicts the working of V3DOFR.

Algorithm 3.1. *Proposed V3DOFR – Voxel based 3D Occlusion Invariant Face Recognition*

Input: *3D Mesh Face Image (I)*

Output: *Classification of Image I*

1. *Pre-processing of Mesh*
    a. Count traingular units of mesh (C_t)
    b. If C_t = 0 do
    c. Voxel = "*Error: No triangular unit found*"
    d. End if
2. *Voxelization according to grid sizes of 4^3, 8^3, and 16^3*
    a. Find the voxel for each triangular unit. Function: *V(size)*
    b. For t = 1 to C_t do
    c. Using three coordinates *A(x1,y1,z1), B(x2,y2,z2),* and *C(x3,y3,z3)*
    d. X = (x1+x2+x3)/3
    e. Y = (y1+y2+y3)/3
    f. Z = (z1+z2+z3)/3
    g. Save V_t = *[X,Y,Z]*
    h. End for
    i. VoxelSize = [4, 8, 16];
    j. C_{v4} = [], C_{v8} = [], C_{v16} = []; *//Current Voxels for all sizes*
    k. C_{v4} = *V*(4);
    l. C_{v8} = *V*(8);
    m. C_{v16} = *V*(16);
3. *Calculation of LPP embeddings*
    a. LPP4 = *LPP*(C_v, 64); *//LPP embedding of grid size 4^3*
    b. LPP8 = *LPP*(C_v, 128); *//LPP embedding of grid size 8^3*
    c. LPP16 = *LPP*(C_v, 256); *//LPP embedding of grid size 16^3*
    d. TotalLPP4 = [TotalLPP4;LPP4;]; *//LPP embedding of grid size 4^3 for all the dataset*
    e. TotalLPP8 = [TotalLPP8;LPP8;]; *//LPP embedding of grid size 8^3 for all the dataset*
    f. TotalLPP16 = [TotalLPP16;LPP16;]; *//LPP embedding of grid size 16^3 for all the dataset*
    g. TotalLPP = [TotalLPP4, TotalLPP8, TotalLPP16];
4. *Triplet Generation with Generator and Discriminator*
    a. Loss = [0,0,0]
    b. NormLoss = [0,0,0]
    c. For t = 1 to 3 do
    d. Select embedding for *A* and *P* randomly from TotalLPP[*t*] of same class and *N* randomly from other class
    e. Adversarial voxel triplet loss training
    f. Loss[*t*] = triplet loss after training //Logarithmic Loss value
    g. End for
5. *Simulated Annealing based Prediction*
    a. FinalLoss = (Loss[1] + Loss[2] + Loss[3])/3;
    b. Try
    c. *If*($e^{FinalLoss} < rand(0,0.8)$)) *//Concept based on simulated annealing*
        //Accept
        Throw*(I_{ID})* *//ID of the image*
    d. Else
        //Reject
        Goto: Step *4* *//Go back to Generator for Triplet Generation*
    e. End If
    f. End try
    g. Catch(I_{ID})
        ID = *Max* (*Corr(I_{ID})*) among all classes

FinalClass = *ID*

h. End catch

The time complexity of the proposed algorithm is as follows. The pre-processing of mesh requires *O(n)* time. In voxelization (i.e. *2(a)-2(h))*, all steps require *O(n)* time and sub-steps (i.e. *2(i)-2(m))* requires *O(1)* time. The calculation of LPP embeddings (i.e. *3(a)-3(c))* requires $O(n^3)$ time [35], and other sub-steps (i.e. *3(d)-3(g))* requires *O(1)* time. Triplet generation with generator and discriminator takes *O(1)* time for steps (i.e. *4(a)-4(b))* and $O(n^3)$ time [181] for rest of the sub-steps (i.e. *4(c)-4(g))*. The simulated annealing based prediction takes *O(1)* time. Hence, the total complexity of proposed technique is $O(n^3)$.

3.3 Experimental Results and Discussion

In this section, the performance of the proposed technique is compared with the existing techniques along with their visual verification. This section presents datasets used, parameter setting, and computational time analysis.

3.3.1 Datasets Used

The datasets used for evaluation of the proposed techniques are Bosphorus face database [153], UMBDB face database [171], and KinectFaceDB face database [164]. Bosphorus dataset consists of 105 subjects in different poses and occlusions. There are 381 occluded images in Bosphorus dataset. All images are annotated with subject ID, pose, occlusion, and emotion description. The total number of images in Bosphorus dataset is 4666. For UMBDB dataset, there are 1473 different images. 590 images are occluded out of 1473 with a different type of occlusions. The number of subjects in the dataset is 143. The modalities of this dataset are 2D and 3D. In KinectFaceDB dataset, there are a total of 52 subject's data. Three types of modalities are covered in this dataset viz. 2D, 2.5D, and 3D. The total number of images is 936, and 312 images are occluded out of it. Table 3.1 presents the detail description of these datasets. Table 3.2 depicts the occlusion description for these datasets.

Table 3.1. Description of datasets used

Dataset	**Number of Subjects**	**Number of Images**	**Modality**	**Annotation**
Bosphorus	105	4666	2D, 3D	Yes
UMBDB	143	1473	2D, 3D	Yes
KinectFaceDB	52	936	2D, 2.5D, 3D	Yes

Table 3.2. Occlusion description for datasets

Dataset	**Bosphorus**	**KinectFaceDB**	**UMBDB**
Occlusion Type Count	4	3	5
Occlusion Types	Eye, Mouth, Glass, Hair	Eye, Mouth, Paper	Cloth, Glass, Hair, Mouth, Paper
Occluded Images Count	381	312	590

3.3.2 Parameter Setting

The parameters of the proposed approach are mentioned in Table 3.3. In the voxelization process, the grid sizes are kept as 4x4x4, 8x8x8, and 16x16x16, respectively. The corresponding number of neighbours for locality preserving projection are 16, 64, and 128 using the current voxels of the corresponding grid size. K-nearest neighbour along with adjacency weight matrix is assigned for an effective LPP embeddings. The number of epochs are 2700, 1200, and 800 corresponding to various grid sizes in triplet loss training. Adaptive moment (Adam) optimizer [91] is employed in triplet loss training. The alpha value is kept to be 0.2 and mean absolute error is used as a loss parameter. The loss function used in the discriminator is the logarithmic loss function. The batch size is kept to be 30, dropout rate is 0.40, learning rate is 5e-3, and activation function used is the rectified linear unit (ReLU).

Table 3.3. Parameter setting for the compared algorithms

Parameter	**Value**
Proposed V3DOFR	
Voxelization	
Grid sizes	4^3, 8^3, 16^3
Locality preserving projection [35]	
Number of neighbours	16, 64, 128
Number of voxels	Current Voxels (V_C)
Neighbour algorithm	K-nearest neighbour
Weight	Adjacency

Adversarial Voxel Triplet loss training [180]	
Number of epochs	2700, 1200, 800
Optimizer	Adam
Alpha	0.2
Overall Loss	Mean Absolute Error
Discriminator Loss	Log Loss
Batch Size	30
Dropout	0.40
Learning Rate	0.005
Activation	ReLU
ElSayed et al. [182]	
Number of inputs	2
Number of hidden layers	1
Nodes in hidden layer	500
Number of outputs	1
Tan et al. [183]	
Input size	256 x 256
CNN model	ResNet-18
Optimizer	Adam
Weight decay	5×10^{-5}
Initial learning rate	0.01
Liu et al. [135]	
Number of SIFT descriptors	128
Yaw poses	$0^{\circ}, \pm 10°, \pm 20°, ..., \pm 90°$
Activation	tanh

ElSayed et al. [182] used Siamese neural network with (2, 500, 1) model, where 2 are the number of inputs, 500 are the number of nodes in the hidden layer, and giving single output. Tan et al. [183] used ResNet-18 model with a 256x256 grid size of the depth map image. Adam optimizer was used with an initial learning rate of 0.01 and weight decay of 5×10^{-5}. Liu et al. [135] built a face reconstruction model based on the pose and expression normalization using 128 SIFT descriptors and tanh activation function for yaw poses of $0^{\circ}, \pm 10°, \pm 20°, ..., \pm 90°$.

3.3.3 Non-Adversarial versus Adversarial Voxel Triplet Generator Face Recognition Technique

This section compares the techniques of 3D face recognition by using adversarial voxel triplet generator and without using the adversarial technique. Table 3.4 shows the performance comparison between non-adversarial and adversarial based voxel triplet generator.

Table 3.4. Comparison between proposed adversarial and non-adversarial based voxel triplet generator face recognition

Dataset	Accuracy		Sensitivity		Specificity	
	Adversarial voxel triplet generator	**Non-adversarial voxel triplet generator**	**Adversarial voxel triplet generator**	**Non-adversarial voxel triplet generator**	**Adversarial voxel triplet generator**	**Non-adversarial voxel triplet generator**
Bosphorus [153]	90.8	82.7	95.8	94.7	70.0	32.7
UMBDB [171]	81.9	75.4	88.5	90.2	43.8	44.5
Kinect Face DB [164]	85.6	77.2	92.7	92.4	45.4	34.8

The accuracy obtained over three datasets is 8-10% better in case of adversarial voxel-triplet based face recognition than the non-adversarial voxel-triplet generator based face recognition. Hence, the adversarial technique in conjunction with simulated annealing has proven to be beneficial for the face recognition accuracy.

3.3.4 Performance Evaluation and Visual Verification of Face Recognition

In this sub-section, the performance of the proposed techniques and four well-known techniques namely ElSayed et al. [182], Tan et al. [183], Liu et al. [135], and Sharma and Kumar method (mentioned in Chapter 4) [87] has been evaluated in four different experimentations. These are 3D face recognition using voxels, under occlusion condition, using landmarks, and using 3D mesh. In each experimentation, the evaluation has been done through seven performance measures viz. Accuracy, Sensitivity, Specificity, Precision, False Positive Rate (FPR), False Negative Ratio (FNR), and F1 Score. The validation of the proposed technique and compared algorithms has been tested over the dataset mentioned in Section 3.1.

Table 3.5. Performance measures obtained from various face recognition techniques using voxels

	Accuracy	**Sensitivity**	**Specificity**	**Precision**	**FPR**	**FNR**	**F1 Score**
Bosphorus Dataset							
ElSayed et al. [182]	89.2	94.5	64.9	90.2	34.3	9.8	90.4
Tan et al. [183]	87.4	95.3	50.2	89.4	54.6	13.1	88.1
Liu et al. [135]	84.7	93.3	31.0	91.0	65.1	7.2	91.9
Sharma	90.0	93.4	**73.7**	94.5	**26.3**	6.6	93.9

and Kumar [87]							
Proposed	**90.8**	**95.8**	70.0	**94.5**	28.4	**6.2**	**94.0**
UMBDB Dataset							
ElSayed et al. [182]	79.2	**91.2**	36.3	87.9	59.2	14.3	87.2
Tan et al. [183]	77.5	89.8	21.4	87.3	58.9	16.0	85.6
Liu et al. [135]	72.8	80.5	**46.0**	80.4	73.7	12.5	83.8
Sharma and Kumar [87]	78.2	89.7	31.6	84.2	68.4	10.3	86.8
Proposed	**81.9**	88.5	43.8	**88.7**	**53.2**	**8.8**	**89.5**
Kinect Face DB							
ElSayed et al. [182]	82.8	**92.9**	27.1	89.5	48.8	10.4	89.6
Tan et al. [183]	79.8	91.9	34.2	86.2	64.9	11.2	87.5
Liu et al. [135]	74.3	84.4	31.3	85.7	57.5	17.5	84.1
Sharma and Kumar [87]	**85.7**	92.2	**50.0**	91.0	50.0	**7.8**	91.6
Proposed	85.6	92.7	45.4	**92.6**	**41.6**	10.3	**92.2**

Table 3.5 shows the performance comparison of various face recognition techniques using voxels. The training dataset has been generated using randomly selected 80% images from the given set. 20% images are used in the testing dataset. In Bosphorus dataset, the proposed technique provides better results than the existing techniques in terms of accuracy sensitivity, precision, FNR, and F1 score. Sharma and Kumar method provides better value of specificity and FPR. The accuracy obtained from the proposed technique is 90.8%. In UMBDB dataset, the proposed technique outperforms the other techniques in terms of performance measures except for sensitivity and specificity. The accuracy achieved by the proposed technique is 81.9%. The main reason behind to drop the accuracy in UMBDB dataset is that the presence of more dynamic occlusion present in UMBDB dataset as compared to Bosphorus dataset. In case of Kinect Face DB dataset, the best accuracy obtained from [87] is 85.6%. However, the proposed technique provides accuracy with a difference of 0.1%. Sharma and Kumar technique outperforms the others in terms of FNR and specificity. Whereas, precision, FPR, and F1 score obtained from the proposed technique is better than the existing techniques. The proposed technique and ElSayed's method achieved the sensitivity of 92.7% and 92.9%, respectively.

Table 3.6. Performance measures obtained from the various face recognition techniques under occlusion condition

	Accuracy	Sensitivity	Specificity	Precision	FPR	FNR	F1 Score
Bosphorus Dataset							
ElSayed et al. [182]	77.5	**93.7**	17.6	**85.3**	66.5	12.4	86.4
Tan et al. [183]	75.6	91.2	42.4	83.9	54.0	15.5	84.2
Liu et al. [135]	73.5	88.2	26.0	82.3	74.1	9.6	86.1
Sharma and Kumar [87]	79.4	90.6	43.6	83.8	56.4	9.4	87.0
Proposed	**81.5**	93.2	**50.2**	84.1	**51.8**	**7.9**	**87.9**
UMBDB Dataset							
ElSayed et al. [182]	64.2	83.9	25.5	72.5	81.3	22.8	78.2
Tan et al. [183]	63.0	**86.7**	19.4	78.3	78.5	26.8	75.7
Liu et al. [135]	60.9	79.1	16.9	79.1	83.5	28.3	75.2
Sharma and Kumar [87]	65.6	74.6	37.7	78.8	62.6	25.4	76.7
Proposed	**67.0**	79.2	**38.0**	**79.9**	**62.3**	**15.2**	**78.6**
Kinect Face DB							
ElSayed et al. [182]	74.6	**91.0**	24.9	80.0	62.6	14.8	82.5
Tan et al. [183]	72.4	90.5	13.6	84.0	63.2	18.4	82.8
Liu et al. [135]	71.3	89.7	19.2	78.0	76.3	12.7	82.4
Sharma and Kumar [87]	76.7	85.9	39.1	83.9	58.9	14.1	85.1
Proposed	**77.9**	88.1	**40.4**	**85.0**	**56.2**	**9.6**	**85.4**

Table 3.6 presents the results obtained from various face recognition techniques under occlusion environment. The proposed model and four techniques were trained with the non-occluded images present in the dataset. However, these were tested on the occluded images. In Bosphorus dataset, the best accuracy obtained from the proposed technique is 81.5%. The accuracy achieved by the proposed approach is better than the second best technique by 2.1%. In terms of sensitivity, the proposed technique is the second best technique. For specificity, FPR, FNR, and F1 Score, the proposed method outperforms the other face recognition methods. In case of precision, the proposed technique and ElSayed et al. provide 84.1% and 85.3% value, respectively. In case of

UMBDB and Kinect Face DB dataset, the proposed technique attained best value for all the performance measures except sensitivity. The best accuracies obtained from the proposed approach over UMBDB and Kinect Face DB are 67% and 77.9%, respectively. The sensitivity and specificity values obtained from the proposed technique are 79.2% and 38.0% for UMBDB, and in case of Kinect Face DB are 88.1% and 40.4%, respectively.

Table 3.7. Performance comparison between different 3D face recognition techniques using landmarks

	Accuracy	**Sensitivity**	**Specificity**	**Precision**	**FPR**	**FNR**	**F1 Score**
Bosphorus Dataset							
ElSayed et al. [182]	83.4	90.6	28.8	90.4	46.2	9.9	90.2
Tan et al. [183]	81.7	91.5	18.9	87.3	69.4	8.5	89.4
Liu et al. [135]	77.5	85.8	55.8	90.3	67.0	13.8	88.2
Sharma and Kumar [87]	84.6	92.4	28.5	88.4	41.5	8.3	90.3
Proposed	**84.9**	**93.8**	**49.7**	**90.8**	**36.6**	**5.6**	**91.3**
UMBDB Dataset							
ElSayed et al. [182]	73.7	84.6	30.1	80.2	62.3	14.2	82.9
Tan et al. [183]	72.3	83.7	34.8	81.3	78.3	14.8	83.2
Liu et al. [135]	69.8	81.9	25.9	75.5	66.6	13.2	80.8
Sharma and Kumar [87]	75.0	89.2	**35.7**	79.5	65.3	12.8	84.0
Proposed	**77.4**	**90.1**	29.7	**85.2**	**60.8**	**10.2**	**85.0**
Kinect Face DB							
ElSayed et al. [182]	79.4	85.2	15.1	80.5	76.3	10.3	84.9
Tan et al. [183]	77.1	83.3	27.6	88.0	61.0	16.8	85.5
Liu et al. [135]	73.9	81.2	36.4	82.5	54.6	17.5	82.5
Sharma and Kumar [87]	80.1	90.0	**37.9**	86.1	62.1	11.2	88.0
Proposed	**81.6**	**91.7**	19.2	**88.7**	**45.9**	**10.0**	**88.8**

Table 3.7 shows the results obtained from 3D face recognition techniques using landmarks. The training and testing datasets are generated in ratio of 80-20 randomly using 26 landmarks in each

case. In case of Bosphorus dataset, the proposed technique proposed best results for most of the performance measures. The highest accuracy achieved from the proposed face recognition technique is 84.9% with 93.8% sensitivity and 49.7% specificity. The proposed technique achieved precision as 90.8%, FPR as 36.6%, FNR as 8.3%, and F1 score as 91.3%. Sharma and Kumar provides the second best performance in terms of accuracy, sensitivity, FPR, FNR, and F1 score. In case of UMBDB dataset, the recognition accuracy obtained from the proposed approach is 77.4%. The proposed method outperforms the other methods in terms of performance measures except specificity. Sharma and Kumar method provides better results than the proposed method in terms of specificity. In case of Kinect Face DB dataset, the proposed technique outperforms all the other methods in terms of evaluation metrics except specificity. Sharma and Kumar method provides more specificity than the proposed technique. The accuracy achieved from the proposed face recognition for Kinect Face DB is 81.6%.

Table 3.8. Performance comparison between different face recognition techniques using 3D mesh

	Accuracy	**Sensitivity**	**Specificity**	**Precision**	**FPR**	**FNR**	**F1 Score**
Bosphorus Dataset							
ElSayed et al. [182]	87.1	93.5	63.6	91.1	34.6	7.3	92.7
Tan et al. [183]	85.9	**94.1**	49.2	91.6	35.4	6.0	92.9
Liu et al. [135]	81.3	90.1	32.4	87.7	58.7	8.9	89.3
Sharma and Kumar [87]	87.4	93.8	38.3	92.2	61.8	6.24	92.9
Proposed	**88.7**	92.8	**68.0**	**92.8**	**30.3**	**5.8**	**93.4**
UMBDB Dataset							
ElSayed et al. [182]	78.4	**90.4**	34.4	83.0	49.9	13.4	86.4
Tan et al. [183]	75.2	88.5	20.5	84.0	60.7	13.0	86.7
Liu et al. [135]	70.6	79.3	**45.8**	78.3	65.0	12.5	82.7
Sharma and Kumar [87]	77.4	89.5	34.9	82.8	65.1	10.5	86.0
Proposed	**79.2**	85.2	42.7	**86.6**	**48.7**	**10.4**	**87.9**
Kinect Face DB							
ElSayed et al. [182]	80.7	89.0	25.8	86.6	48.8	8.7	89.7
Tan et al. [183]	77.5	90.7	33.7	88.9	45.7	14.9	86.6

Liu et al. [135]	72.3	82.1	30.2	82.4	56.7	9.6	86.2
Sharma and Kumar [87]	**85.7**	**92.1**	**58.4**	**90.4**	**41.6**	**7.9**	**91.2**
Proposed	83.7	90.1	42.3	87.5	51.4	12.9	87.3

Table 3.8 shows the results obtained from different face recognition techniques using mesh. The training and testing dataset is partitioned into 80-20 ratio randomly. For Bosphorus dataset, the accuracy achieved from the proposed face recognition technique is 88.7%. While, Sharma and Kumar method provides the accuracy of 87.4%. The sensitivity achieved by the proposed technique is 92.8%. Tan's method attained the best sensitivity of 94.1%. The proposed technique achieved the best results in terms of accuracy. In case of UMBDB dataset, the best accuracy for 3D face recognition obtained from the proposed technique is 79.2%. The best sensitivity achieved by ElSayed's method with 5.2% difference from the proposed technique. The precision, FPR, FNR, and F1 score of the proposed technique are 86.6%, 48.7%, 10.4%, and 87.9%, respectively. Liu's method provides better specificity of 45.8% for UMBDB dataset. In case of Kinect Face DB, Sharma and Kumar outperforms the other recognition methods for all evaluation metrics.

3.3.5 Visual Verification

Figure 3.6 shows the visual verification of random 3D mesh images based on occlusion invariant proposed framework. All 3D meshes have an occlusion viz. hand on eyes, hair, glasses, hands-on mouth, cloth, cap, and finger. Ten 3D meshes from the occluded images are selected randomly for verification. The predicted subject IDs are given along with actual subject IDs. Nine out of ten meshes have the correct predicted value during verification. Hence, it validates the proposed method is an occlusion invariant.

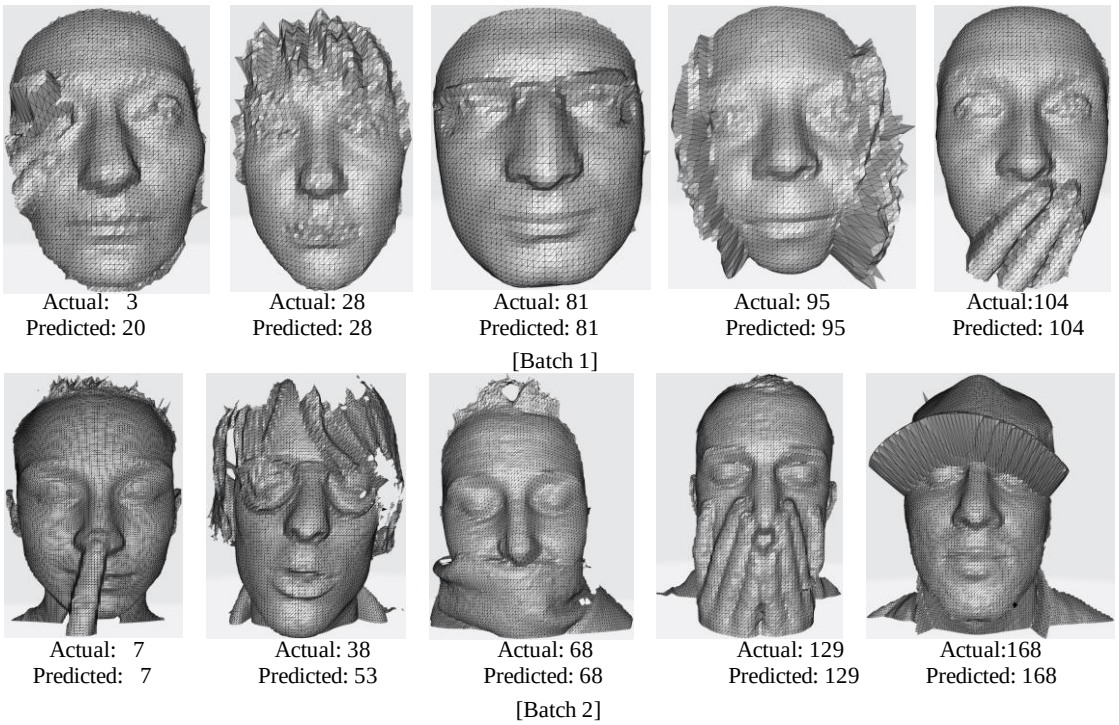


Figure 3.6. Visual verification of 3D meshes with their actual and predicted subject IDs in Batch 1 and Batch 2

3.3.6 Computational Time Analysis

Table 3.9 depicts GPU based computational time obtained from the proposed approach and other techniques. Four well-known 3D face recognition techniques are compared with the proposed method using voxels, landmarks, and meshes. The computation time is presented in Table 3.9. The computation time is the average time in all the phases. The experimentation has been done on GeForce GTX 1080Ti GPU model with 3584 Compute Unified Device Architecture (CUDA) cores and a memory speed of 11 Gbps.

Table 3.9. Computation time (in ms) on GPU for proposed technique versus other techniques in voxel-based face recognition

Technique	**Pre-Processing**	**Recognition**	**Verification**	**Learning Model**
ElSayed et al. [182]	90	81	107	Siamese neural network
Tan et al. [183]	87	88	104	Convolutional neural network
Liu et al. [135]	96	73	112	Pose and expression normalized
Sharma and Kumar [87]	78	66	98	VAE, BiLSTM
Proposed using voxels	**80**	**65**	**99**	Adversarial triplet loss
Proposed using	**55**	**67**	**95**	Adversarial triplet loss

landmarks				
Proposed using meshes	**105**	**69**	**105**	Adversarial triplet loss

The overall computation time of the proposed technique is the fastest using landmarks, and slowest using meshes.

3.3.7 Convergence Analysis

Figure 3.7 shows the convergence of accuracy obtained from the proposed technique for all datasets. Bosphorus dataset converges at 90% accuracy in 2700 epochs, UMBDB dataset converges at 81% accuracy in 1200 epochs, and KinectFaceDB dataset converges at 85% in 800 epochs. The convergence plot shows the accuracy obtained from combined approaches of triplet loss training, simulated annealing, and game theory.

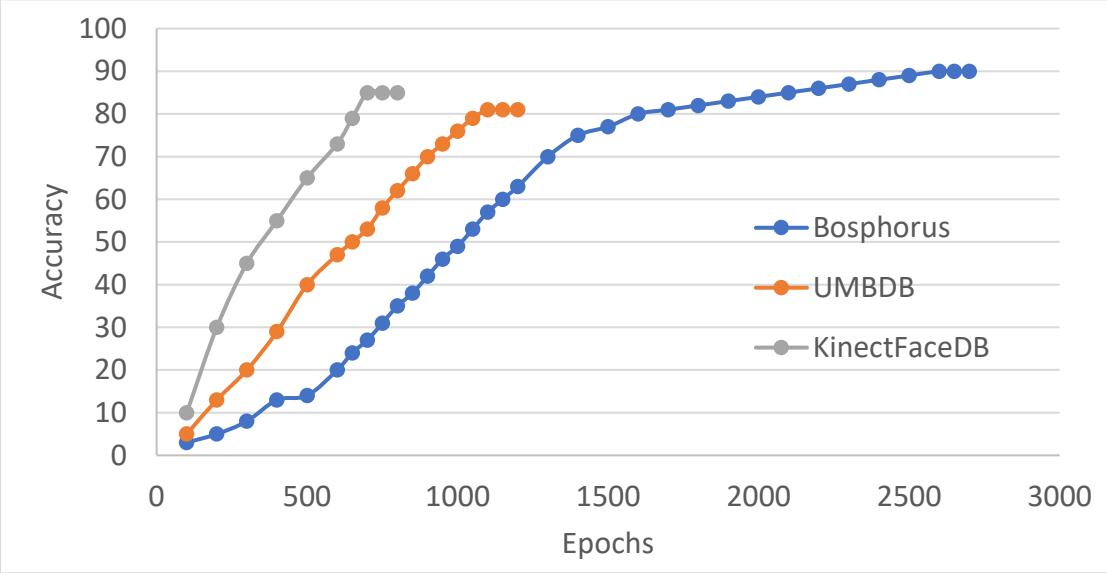


Figure 3.7. Accuracy based convergence plot for the proposed approach

Figure 3.8 shows the accuracies obtained from the face recognition model using voxelized technique over three datasets viz. Bosphorus, UMBDB, and KinectFaceDB. It can be seen from Figure 3.8 that the accuracy on Bosphorus dataset is 90.8%, UMBDB is 81.9%, and KinectFaceDB dataset is 85.6%.

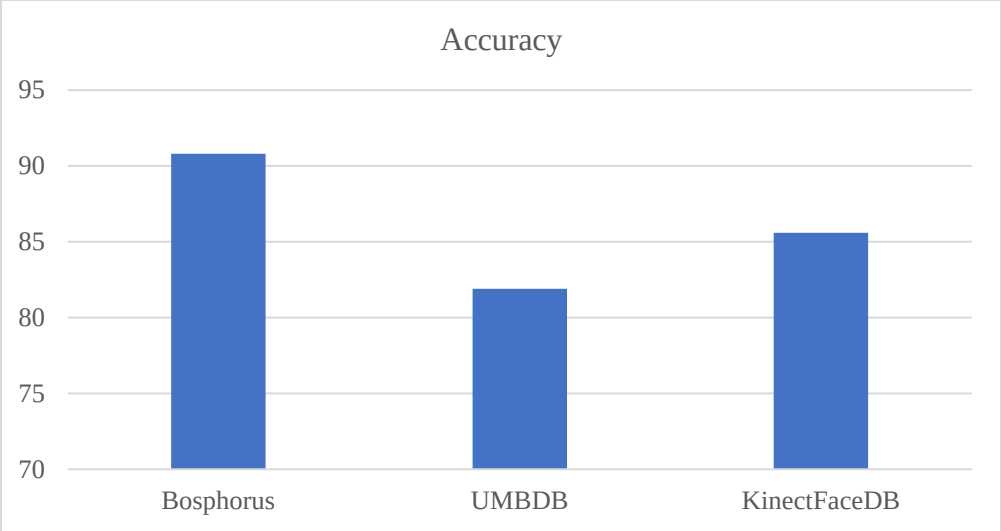


Figure 3.8. Accuracy comparison of three datasets using the proposed approach

3.4 Summary

In this chapter, voxel-based 3D occlusion invariant face recognition framework is proposed. The proposed framework utilizes the concept of generator and discriminator. Bosphorus, UMBDB, and Kinect Face DB have been used for implementing face recognition techniques. The best average accuracy obtained from voxel-based face recognition is 86.1%. For occlusion invariant face recognition, the average given by the proposed technique is 75.5%. In case of face recognition using 3D landmarks, the best average accuracy obtained from the proposed technique is 81.3%. In case of face recognition using 3D meshes, the average accuracy obtained from the proposed technique is 83.9%. The adversarial training strategy for triplet generation ensures low biasness. The technique coupled with simulated annealing allows the proposed method to be robust in different areas using voxels.

4. Voxel-based 3D Face Reconstruction Using Sequential Deep Learning

4.1 Introduction

This chapter presents a novel 3D face reconstruction technique is proposed along with a sequential deep learning-based framework for face recognition. It uses the voxels generated from the voxelization process. It uses the reflection principle for generating reconstructed point in 3D using the mid-face plane. From the reconstructed face, a sequential deep learning framework is developed to recognize gender, emotion, occlusion, and person. The developed framework utilizes the concepts of variational autoencoders, bidirectional long short-term memory, and triplet loss training. The sequential deep learning model extracts and refines the reconstructed voxels by generating deep features. The support vector machine is applied to deep features for final prediction. The proposed 3D face recognition system is compared with three well-known deep learning approaches over three occluded datasets.

4.2 Proposed 3D Face Recognition Framework

This section describes the motivation behind the 3D face recognition followed by the proposed framework.

4.2.1 Motivation

The main challenges of 3D face recognition are the number of voxels may vary and high time complexity of voxel-based technique. To resolve the former problem, fuzzy c-means clustering is used. To tackle the latter one, deep features have been used. The proposed framework is inspired from the concept of mirroring image [184]–[186].

The existing techniques cannot extract the deep features of 3D face voxels. Variational autoencoder based framework is proposed that extract deep features from the input voxels. The deep features help Bidirectional LSTM along with triplet loss training to generate consistent embedding for better classification.

The human face is nearly symmetrical along the vertical axis of nose tip. This concept is leveraged

and the random sample of eight images is visually verified in Figure 4.1. After the selection of the frontal pose face images, each image is cropped from the left side of face. The mirror image of the matrix of pixels is found using the mid-face plane. The mirrored images generated are tested for similarity with the original image using the concept of correlation. The higher correlation value means that the more similar is the original image to the mirror image.

4.2.2 Proposed 3D Face Recognition Framework

There are two phases of 3D face recognition namely, training and testing phase. In the proposed architecture, the training phase is divided into four parts. These are preprocessing, reconstruction, deep learning, and prediction. In pre-processing, a 3D mesh image is voxelized in three different sizes of 8^3, 16^3, and 32^3 using the voxelization technique. The voxelization of different pose mesh images produces a different number of voxels intra-class voxels. Voxel clustering is performed to handle this sparsity of the number of voxels in intra-class voxelization. Reconstruction is performed on the clustered forms of voxels by using 3D voxel-based face reconstruction technique (*3DVFR*).

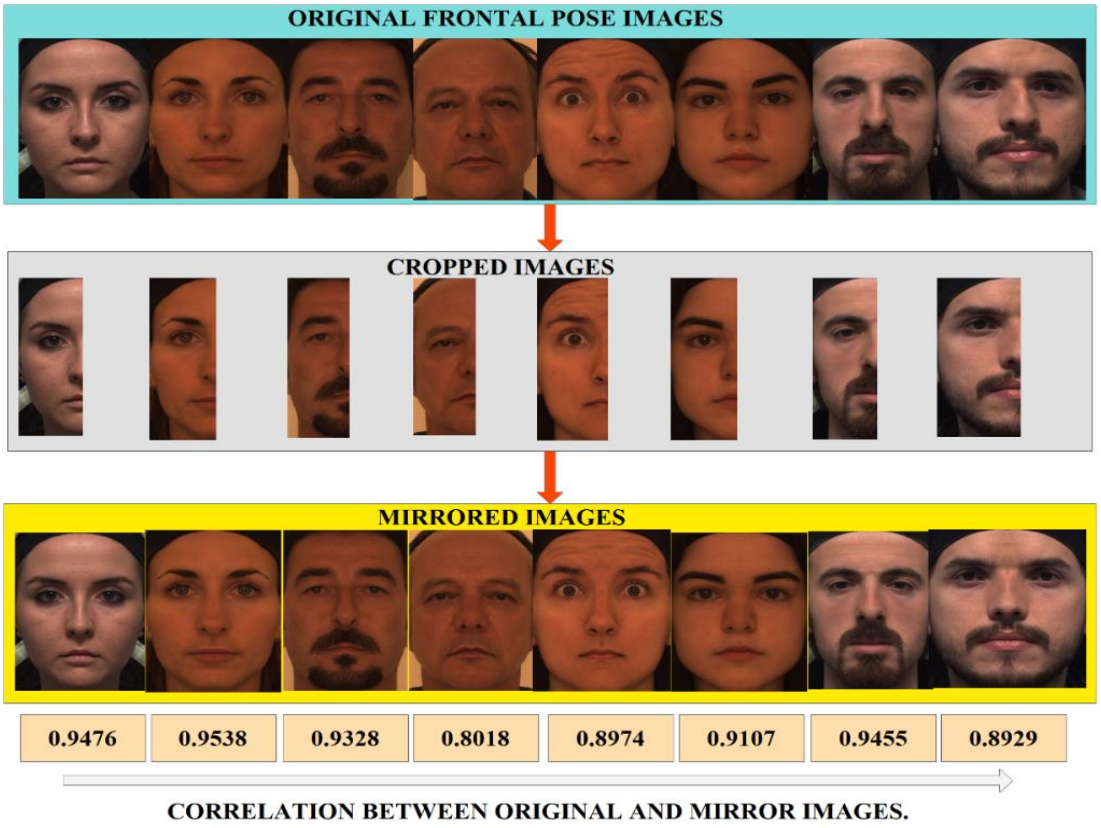


Figure 4.1. Mirroring of face images

After the reconstruction, deep learning is performed using the final reconstructed landmarks/voxels for deep feature embeddings. There are three deep learning techniques used in succession for achieving these embeddings. These are variational autoencoders, a variant of recurrent neural network viz. bidirectional long short-term memory model. The triplet loss training is applied on final embedding for classification. The final prediction of face class is achieved through support vector machine (SVM).

During the testing phase of the proposed architecture, two standard methods are applied viz. validation and verification. The final trained model is used to achieve these two testing methods. The prediction value for verification and validation are compared with the ground truth values to produce overall accuracy. Figure 4.2 presents the proposed framework for 3D face recognition using voxel reconstruction. The detail description of sub-phases is given in the preceding subsections.

4.2.2.1 Pre-processing

3D mesh images cannot be directly used for reconstruction. The voxels are extracted from the mesh using voxelization process. In a particular region, if the voxel exists then the location is given true value else it is given false value. The number of voxels generated from 3D mesh depends on the size of mesh given as input. When multiple mesh images are processed for voxels, the number of voxels generated is not consistent and needs preprocessing in terms of removing the sparsity of the array. Fuzzy c-means (*FCM*) clustering [187] plays an important role in bringing all the voxels count into the same size. This algorithm assigns membership to each voxel corresponding to each cluster based on the cluster center and voxel point. Voxel's membership for a particular cluster depend upon its distance from center to cluster.

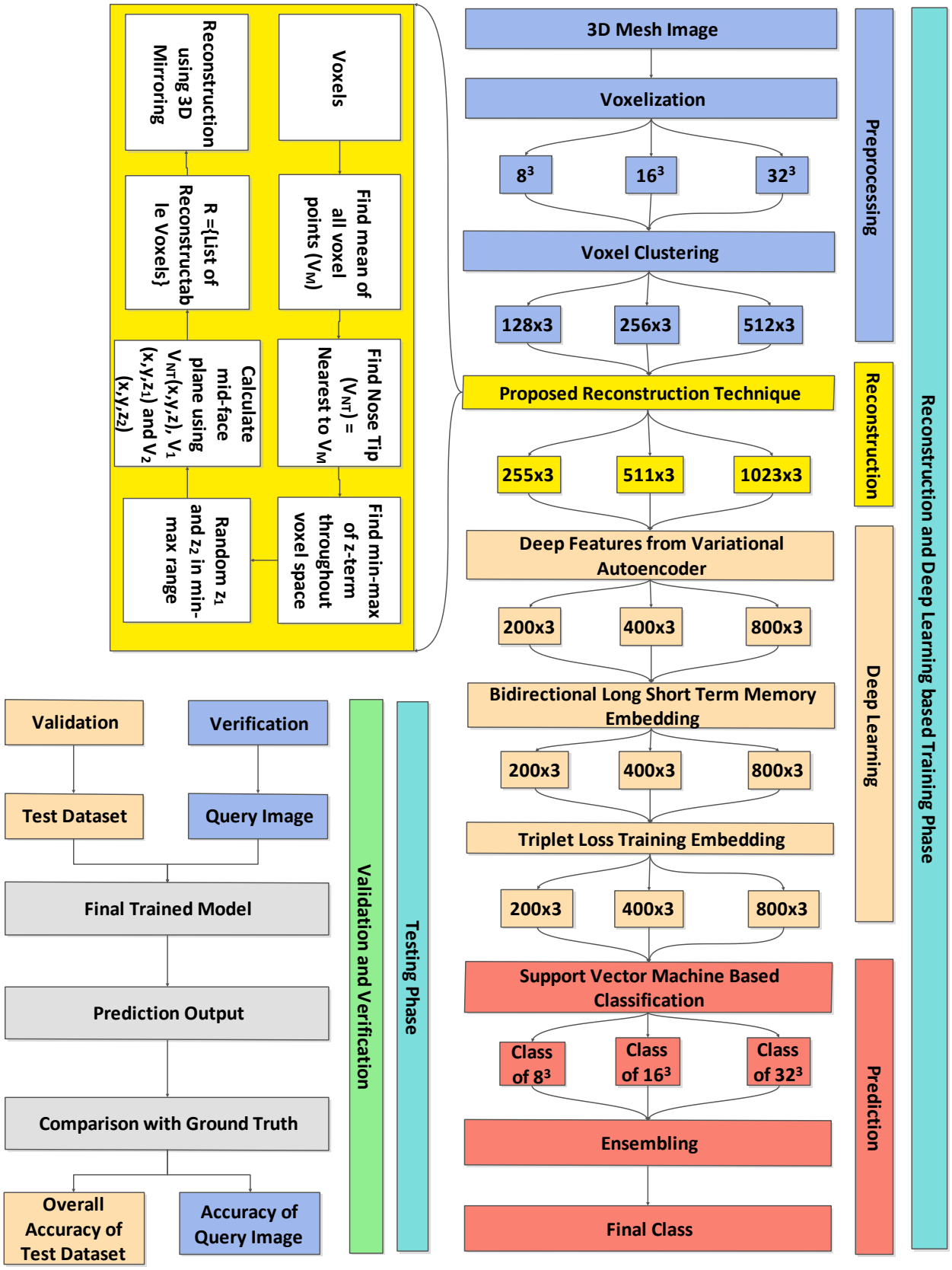


Figure 4.2. Proposed Framework for 3D Face Recognition using Voxel Reconstruction

FCM is based on the concept of minimization of the objective function, which is given below [187]:

$$J_m = \sum_{i=1}^{N}\sum_{j=1}^{c} \mu_{ij}^{m} \| x_i - c_j \|^2, \ 1 \le \mathrm{m} < \infty \tag{4.6}$$

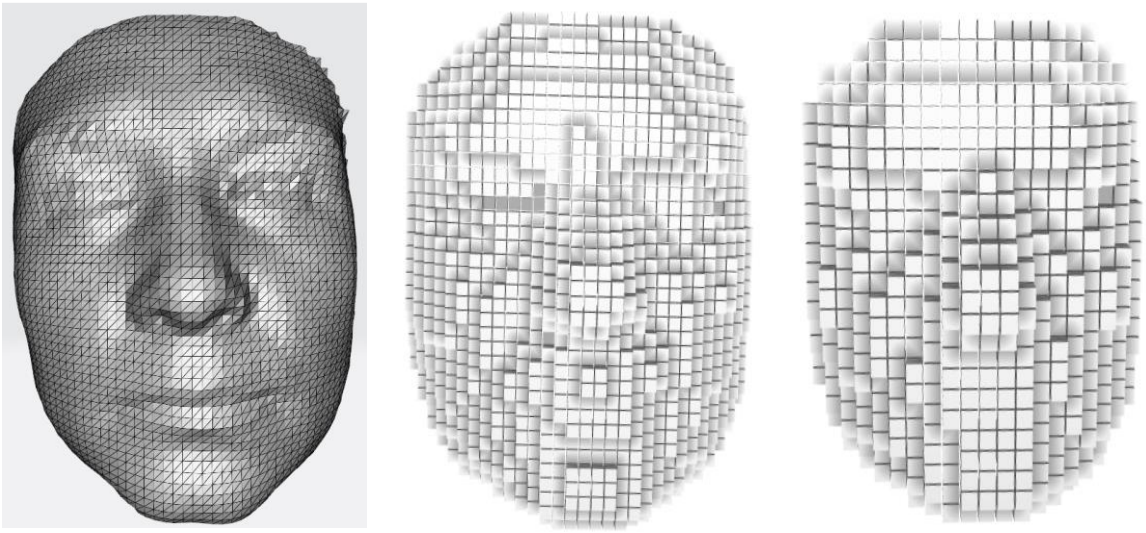
where $m>1$ and m belongs to real numbers, μ_{ij} is the membership degree of x_i in cluster j. x_i is i^{th} data point, c_j is the d-dimension center of cluster and ||*|| denotes the similarity between the cluster center and the data. μ_{ij} is computed as follows [187]:

$$\mu_{ij} = \left(\sum_{k=1}^{c} \left(\frac{\| x_i - c_j \|}{\| x_i - c_k \|} \right)^{\frac{2}{m-1}} \right)^{-1} \tag{4.7}$$

Where c_j can be defined as [187]:

$$c_j = \frac{\sum_{i=1}^{N} \mu_{ij}^{m} x_i}{\sum_{i=1}^{N} \mu_{ij}^{m}} \tag{4.8}$$

This iteration will terminate when $\max_{ij}\{| \mu_{ij}^{k+1} - \mu_{ij}^{k} |\} < \in$, where k denotes iteration steps and $\in = (0, 1)$. The convergence of this whole procedure is to a local minimum J_m. Figure 4.3 presents the visualization of preprocessing of voxelization and clustering of voxels.



A. 3D Mesh Image B. Voxelization C. Clustered Voxels

Figure 4.3. Preprocessing of 3D Mesh Image into Voxelization and Clustering of Voxels

4.2.2.2 Reconstruction

This subsection discusses the reconstruction of the face using a mirror image technique. A 3D voxel-based face reconstruction (3DVFR) technique is proposed. It involves pre-processing the available frontal mesh image for voxelization and reconstruction of the voxels based on the available information.

Algorithm 4.1. *3DVFR – 3D Voxel-based Face Reconstruction Technique*

Input: *Dataset D is divided into Train (T_r) and Test (T_s) sets*
T_r = All mesh images without any kind of rotation of face.
T_s = All mesh images having some kind of rotation of face.
Mesh image I_m taken as input from T_r.
Output: *Reconstructed 3D voxels*

1. Count triangular units of mesh (*C*).
2. *If C=0 do*
   Voxel = "Error: Cannot be generated"
   End if
3. Finding the voxel for each *C*

a. *For t = 1 to C do*
   b. Using triangle *t* coordinates, find centroid O(C_x, C_y, C_z)
   c. $x = C_x$
   d. $y = C_y$
   e. $z = C_z$
   f. save Voxel = [*x*, *y*, *z*]

g. *End for*

4. Derive voxelized image (*V*) from the voxels generated in step *3*.
5. Apply fuzzy c-mean clustering on *V* to extract *N* voxels based on fine to coarse details.
6. Find mean voxel (V_M)

   $V_M = \frac{1}{N}\sum_{i=1}^{N} V_i$, where *N* = Number of voxels point clusters.

7. Calculate Euclidean square distance (D_i) of each voxel V_i from V_M and find Nose Tip (V_{NT}):

   $V_{NT} = Min\ (D_i)\ \forall\ V_i$, where $i \in [1, N]$.

8. Find maximum and minimum *z* coordinate (z_{min} and z_{max}) from all the voxels [V_1,V_2,…, V_C].
9. Choose two random *z* values (z_1 and z_2) between z_{min} and z_{max}.
10. Using $V_{NT}(x,y,z)$, $V_1(x,y,z_1)$, and $V_2(x,y,z_2)$, calculate 3D mid-face plane.
11. Find the array (*R*) of all reconstructable points on the face.

R = {set of all voxels not lying on mid-face plane}

12. Reconstruct the 3D face voxels using the concept of mirroring through mid-face plane for all the R number of voxels on the face.

a. For t = 1 to len(R) do

b. $P = R[t] = [x_1, y_1, z_1]$ *//P is reconstructable 3D point*

c. $N = [x_2, y_2, z_2]$ *//PN is perpendicular to mid-face plane*

d. *PN* direction ratios = *[a, b, c] //a, b, c are constants*

e. $PN = [(x-x_1)/a = (y-y_1)/b = (z-z_1)/c = k]$ *//k is constant*

f. $x = ak + x_1, y = bk + y_1, z = ck + z_1$

g. Since *N* lies on mid-face plane

h. $k=(-ax_1-by_1-cz_1-d)/(a^2+b^2+c^2)$

i. $x_2=ak+x_1, y_2=bk+y_1, z_2=ck+z_1$

j. return $[2x_2-x_1, 2y_2-y_1, 2z_2-z_1]$ *//Reconstructed 3D Voxel*

k. End for

The dataset is divided into training (T_r) and testing (T_s) sets based on non-yaw rotation and yaw rotation, respectively. The input of one of the mesh image (I_m) is selected from T_r. The mesh image is selected as the triangular meshes. It is converted into voxel form for further processing (see Steps 1-4). All the triangular units are counted and based on centroid information. The voxels are generated for the mesh. The resultant image is considered a voxelized image (V). The number of voxels is generated by varying the pose of the face images. To normalize the number of voxels, FCM clustering is used (Step 5). The number of voxels generated using this technique helps in normalization of the dataset.

3D mid-face plane is calculated for reconstruction of missing voxels along with the array of reconstructable points is generated. The training set images hold all the images with frontal pose only. Hence, the mean voxel is calculated over the face as the nose tip voxel point cluster. Using Euclidean square distance, Nose Tip (V_{NT}) is calculated over the voxel clusters space. The minimum (z_{min}) and maximum (z_{max}) value of z-ent is calculated over all the voxel clusters for generating a mid-face plane. Two values of z-element (z_1 and z_2) are selected at random in the range from z_{min} to z_{max}. Using $V_{NT}(x,y,z)$ and two other 3D points $V1(x,y,z_1)$, $V2(x,y,z_2)$, mid-face plane is generated as Eq. 4.9 $ax + by + cz + k = 0$. Array (R) consists of all voxels that are eligible for reconstruction, those voxel points are not lying on the mid-face plane (see Steps 6 to 11).

After the mid-face plane is found, the list of reconstructable voxels is calculated using all the voxels in the current space and checking if voxel is lying on the mid-face plane or not. Those voxels do not lie on the plane are added to the list of reconstructable voxels. Reconstruction is done using the concept of mirror image. For all reconstructable points R, the corresponding 3D points are on the other side of the mid-face plane is computed. Let *P(x1, y1, z1)* be 3D point whose reflection is to be calculated and *N(x2, y2, z2)* be 3D point on the mid-face plane such that P and N form 90° with respect to mid-face plane. With a, b, and c as the direction ratios, PN is defined as:

$$\frac{x - x_1}{a} = \frac{y - y_1}{b} = \frac{z - z_1}{c} = k \tag{4.10}$$

Since N lies on the mid-face plane, according to the rules of 3D coordinate geometry,

$$k = \frac{-(ax_1 + by_1 + cz_1 + d)}{a^2 + b^2 + c^2} \tag{4.11}$$

So, $x_2 = ak + x_1, y_2 = bk + y_1$ and $z_2 = ck + z_1$. (4.12)

When the coordinates of N are known, then reflected point is calculated as:

$$I_{Rf} = (2x_2 - x_1, 2y_2 - y_1, 2z_2 - z_1) \tag{4.13}$$

Hence, if the number of voxels generated after clustering is N, then the total number of reconstructed voxels is *N-1* and the total number of voxels generated after the reconstruction process is approximately *2(N-1) + 1*.

4.2.2.3 Deep Learning

After the reconstruction phase, deep learning is applied to the different sized voxels. The number of voxels nearly gets doubled after using reconstruction on the voxels generated from FCM. Two sequential deep learning models, namely VAE and BiLSTM, are applied along with the triplet loss training. With reference to Figure 4.7, the reconstructed voxels are generated as 255x3, 511x3, and 1023x3 for 8^3,16^3, and 32^3 voxelized 3D mesh images, respectively. The deep features are extracted using variational autoencoder as 200x3, 400x3, and 800x3, respectively. Autoencoders provides an efficient dimensional reduction [188]. VAE provides the deep features in a compressed form as compared to the input by using the probability distribution based latent transformation

[54], [189]. Voxel image is sequential locally for facial features. Hence, BiLSTM further optimizes the deep features by using long term dependencies between the time steps of sequential data for its input into triplet loss training by forming triplets [190]–[192]. Triplet loss training provides the final embedding for prediction phase by simultaneously reducing the distance between anchor and positive embedding of triplet and increasing the distance between anchor and negative embedding. Triplet loss training handles the imbalanced classes efficiently [193].

4.2.2.4 Prediction

The deep learning embedding is given as input to support vector machine-based classification model [194]–[198]. The support vector machine takes 200x3, 400x3, and 800x3 embeddings as input and provides classes of 8^3, 16^3, and 32^3 voxels for each 3D mesh input image. The final step of the prediction is the application of voting based ensemble model [199]–[201] for final class prediction.

Once the model is trained, the validation and verification are done in the testing phase. During validation, the accuracy is calculated for the whole test dataset by comparing the prediction output with ground truth. Verification is valid on a single query image. The trained model is used for the prediction of the query image and the solution is compared with ground truth for prediction accuracy.

4.2.3 Computational Complexity

The input of proposed algorithm is a 3D mesh image selected from the training dataset, which requires *O(n)* time. The triangular units of the mesh *(C)* requires *O(n)* time. The count of triangular units requires *O(1)*. The voxelization process of all the triangular units is done in iteration from the initial to final triangular unit. Therefore, it requires *O(n)*. The voxelized image after voxelization needs *O(1)*. After the voxelization process, FCM clustering requires $O(ndc^2i)$, where *n* is the number of data points, *d* is the number of dimensions, *c* is the number of clusters, and *i* is the number of iterations. Finding the mean voxel is of *O(n)*. The computation of Nose Tip (V_{NT}) requires *O(n)*. The computation of the minimum and maximum z-coordinate values is *O(n)* and choosing two random *z* values from min-max range is *O(1)*. The computation of mid-face plane is

O(1). Finally, the set of reconstructable voxel points and reconstructing with mirroring technique requires *O(n)*.

4.3 Results and Discussion

In this section, the proposed approach for voxel reconstruction is compared with three well-known techniques over 3D face datasets.

4.3.1 Datasets Used

Bosphorus dataset [153] consists of 105 subjects in different occlusions, emotions and poses with a total of 4666 facial samples. Every subject has seven emotional expressions including the neutral and multiple head poses over 0-90 degree. There is a total of 381 samples having four types of occlusions in them. Every sample has a 2D image, a 3D point cloud, and landmark files associated with both of them. UMBDB dataset [171] consists of 143 subjects with a total of 1473 samples of 2D as well as 3D images. There are 590 occluded images of multiple types, such as cloth, glass, hand, bottle, cigarette, and others. KinectFaceDB dataset [164] is an RGBD sensor-based face database of different facial expressions in multiple lighting as well as occlusion conditions. There are 52 subjects in the dataset. Data collection is done in two sessions with a gap of one-two weeks. RGB image, depth image, 3D object file, and landmark files of all three types are available in the dataset. Table 4.1 provides a detailed description of the datasets used.

Table 4.1. Detailed description of datasets

Dataset	**Bosphorus**	**UMBDB**	**KinectFaceDB**
Dimensions	2D, 3D	2D, 3D	2D, 2.5D, 3D
Number of Subjects	105	143	52
Number of Images	4666	1473	936
Number of occluded images	381	590	312
Occlusion Classes	Hair, Glasses, Mouth, Eye	Cloth, Glasses, Hand, Others	Eye, Paper, Mouth
Occlusion Classes Annotation	Yes	No	Yes
Pose Based Occlusion	[-90 to +90]	No	No

Rotation Based Occlusion	Yaw Rotation, Cross Rotation	No	No
Degree of Occlusion	0 – 50 %	0 – 80 %	0 – 50 %
Synthetically	No	No	No
Scanner	Stereo	Laser	Kinect sensor
Condition	Controlled	Controlled	Controlled

4.3.2 Parameter setting for the proposed algorithm

The parameters of the proposed framework are shown in Table 4.2. A 3D mesh image is voxelized into three different sizes of 8^3, 16^3, and 32^3 respectively. Due to the variation in number of voxels generated in the voxelization process, FCM clustering is used to normalize these voxels to form three different number of cluster centers (128, 256, and 512 respectively). Further, these clusters are applied on VAE to generate deep features. Similarly, the output of VAE is applied to BiLSTM for optimization of deep features. The triplet loss training is done by using Adam optimizer, which provides the final embedding for SVM.

Table 4.2. Parameter setting for the proposed approach

Parameter	**Value**
Voxelization	
Sizes	8^3, 16^3, and 32^3 respectively
Fuzzy C-Means Clustering	
Number of cluster centers	128, 256, and 512 respectively
VAE	
Number of Epochs	1200, 1200, and 1100 respectively
Batch Size	100, 50, and 10 respectively
Activation	ReLU
Regularization	L2
Latent Dimension	100, 200, and 400 respectively
Optimizer	Adam
BiLSTM	

Number of Epochs	240, 230, and 220 respectively
Batch Size	100, 50, and 10 respectively
Activation	
- Input Gate	Sigmoid
- Output Gate	Sigmoid
- Forget Gate	Sigmoid
- Cell Output	Tanh
Dropout	0.2, 0.3 and, 0.4 respectively
Optimizer	Adam
Triplet Loss	
Number of Epochs	3050, 2850, and 2700 respectively
Alpha	0.2
Optimizer	Adam
Loss	Mean Absolute Error

4.3.3 Performance Evaluation

The performance evaluation of *3DVFR* reconstruction technique is tested on the above-mentioned datasets for gender, emotion, occlusion, and person recognition before and after reconstruction.

4.3.3.1 Gender Recognition

Table 4.3 depicts the accuracy comparison of three datasets namely Bosphorus, UMBDB, and KinectFaceDB for gender recognition. The accuracy of gender prediction improves by 5.8% after reconstruction for Bosphorus dataset, 8.6% for UMBDB and 8.5% for KinectFaceDB dataset. Firstly, the reason behind the improved accuracy of gender recognition is that after the reconstruction of face, the number of voxels is double. It makes the deep learning based recognition model perform better than the others. Secondly, the accuracy for three datasets is above 90% because there are only two classes such as male and female. The sensitivity and specificity values show decent improvement because of the appropriate classification of true positive and negative values. In case of FPR and FNR, the values have decreased after reconstruction for gender recognition as true positive and negative values have been correctly classified.

Table 4.3. Quantitative analysis of gender recognition using voxels

Metrics Dataset	Accuracy	Sensitivity	Specificity	Precision	FPR	FNR	F1 Score
	Before 3D Face Reconstruction						
Bosphorus	91.449	95.432	55.292	95.092	44.708	4.568	95.262
UMBDB	83.503	92.707	35.983	88.204	64.017	7.293	90.399
KinectFaceDB	85.897	92.761	39.669	91.194	60.331	7.239	91.971
	After 3D Face Reconstruction						
Bosphorus	97.28	98.86	85.02	98.08	14.98	1.14	98.47
UMBDB	92.12	95.34	59.20	96.05	43.80	4.66	95.70
KinectFaceDB	94.44	97.48	59.46	96.55	40.54	2.55	97.00

Figure 4.4 shows the gender-wise accuracy plot for each dataset. The proposed model recognizes male with better accuracy than females. This is primarily due to a smaller number of females in datasets as compared to males. In KinectFaceDB, the accuracy of male prediction is 96.20% and female is 92.68%. In UMBDB dataset, the accuracies for male and female prediction are 95.35% and 88.89%, respectively. In the Bosphorus dataset, the male and female prediction accuracies are 98.56% and 96.00%, respectively.

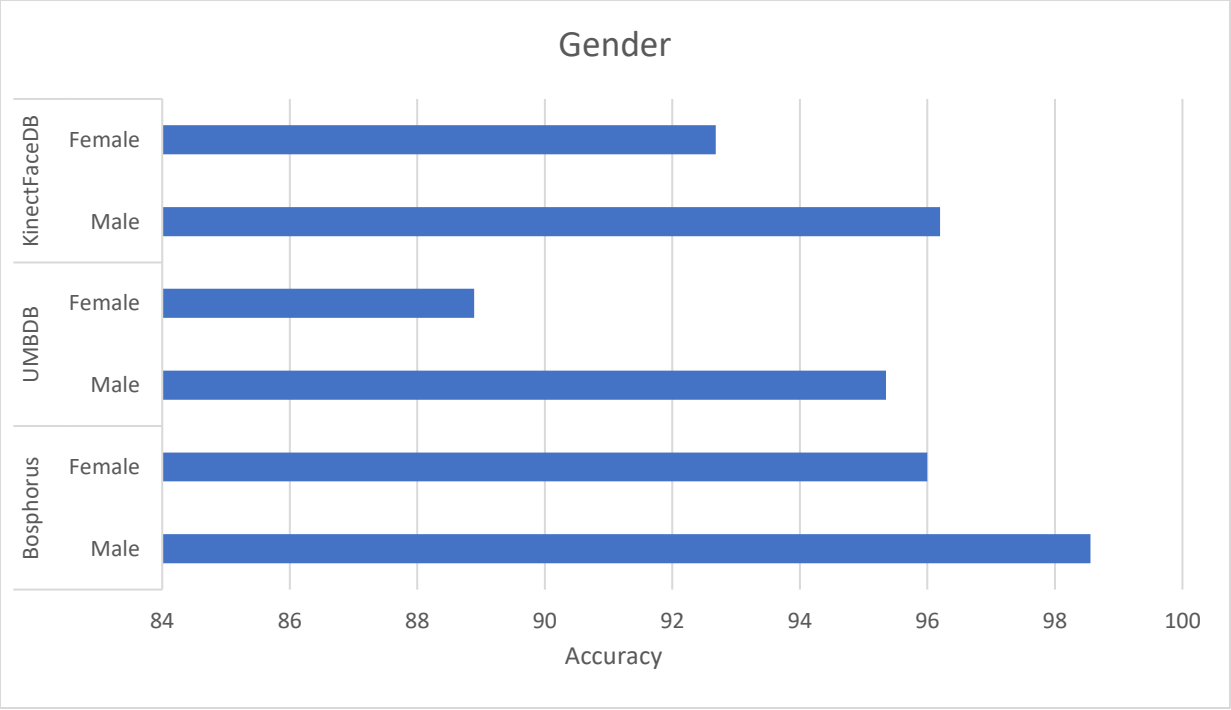


Figure 4.4. Voxel-based accuracy representation for gender recognition

4.3.3.2 Emotion Recognition

Table 4.4 presents a quantitative analysis of emotion recognition. The overall increase in accuracy before and after the reconstruction is 10.62%, 14.9% and 12.49% for Bosphorus, UMBDB and KinectFaceDB datasets, respectively. The primary reason for improved accuracy after reconstruction is the same as in the case of gender recognition, which is the number of voxels are doubled. The accuracy has shown improvement due to less number of classes for all the datasets. In case of FPR and FNR, the values have decreased after reconstruction for emotion recognition. F1 score has improved from 7-10% after the reconstruction.

Table 4.4. Quantitative results of emotion recognition

Metrics **Dataset**	**Accuracy**	**Sensitivity**	**Specificity**	**Precision**	**FPR**	**FNR**	**F1 Score**
	Before 3D Face Reconstruction						
Bosphorus	83.95	94.01	34.64	87.57	65.36	5.99	90.68
UMBDB	72.37	87.26	11.11	80.16	88.89	12.74	83.56
KinectFaceDB	77.46	90.93	23.12	82.67	76.88	9.07	86.60
	After 3D Face Reconstruction						
Bosphorus	94.57	96.89	53.22	97.36	46.77	3.10	97.12
UMBDB	87.78	93.17	49.72	92.89	50.27	6.82	93.03
KinectFaceDB	89.95	94.16	53.12	94.61	46.87	5.83	94.39

Table 4.5 shows the classification of emotions for each dataset. Bosphorus dataset has seven classes of emotions. UMBDB dataset has five different types of emotions, and KinectFaceDB dataset has three categories of emotions.

Table 4.5. Emotions classification for datasets

S.No.	**Dataset**	**Number of emotions**	**Emotion classes**
1.	Bosphorus	7	Anger, Disgust, Fear, Happy, Sadness, Surprise, Neutral.
2.	UMBDB	5	Anger, Disgust, Happy, Sadness, Neutral.
3.	KinectFaceDB	3	Happy, Surprise, Neutral.

Figure 4.5 depicts the qualitative analysis of classification of emotions for three datasets. Neutral emotion is recognized with the highest accuracy, followed by fear and happiness. In case of KinectFaceDB, happy emotion is recognized with an accuracy of 87.6788%, surprise with the accuracy of 85.7581%, and neutral with an accuracy of 96.4130%. In UMBDB dataset, the accuracy of recognition for happy emotion is 88.8061%, sadness is 81.0609%, anger is 87.5489%, disgust is 85.5958, and neutral is 95.8880%, respectively. In Bosphorus dataset, accuracy for happy emotion is 96.6975%, surprise is 91.2190%, sadness is 87.3164%, fear is 97.9006%, anger is 94.2960%, disgust is 95.7788%, and neutral is 98.7815%, respectively. The accuracy of neutral emotion is the highest for all three datasets because the neutral face is the most common emotion amongst all emotions. On the other side, sadness is observed with minimum accuracy as this type of emotion is generally unpredictable. Sadness is considered as a neutral emotion. Fear and happy face are prominent in the real world. Hence, they have shown higher accuracy as compared to sadness and surprise emotions.

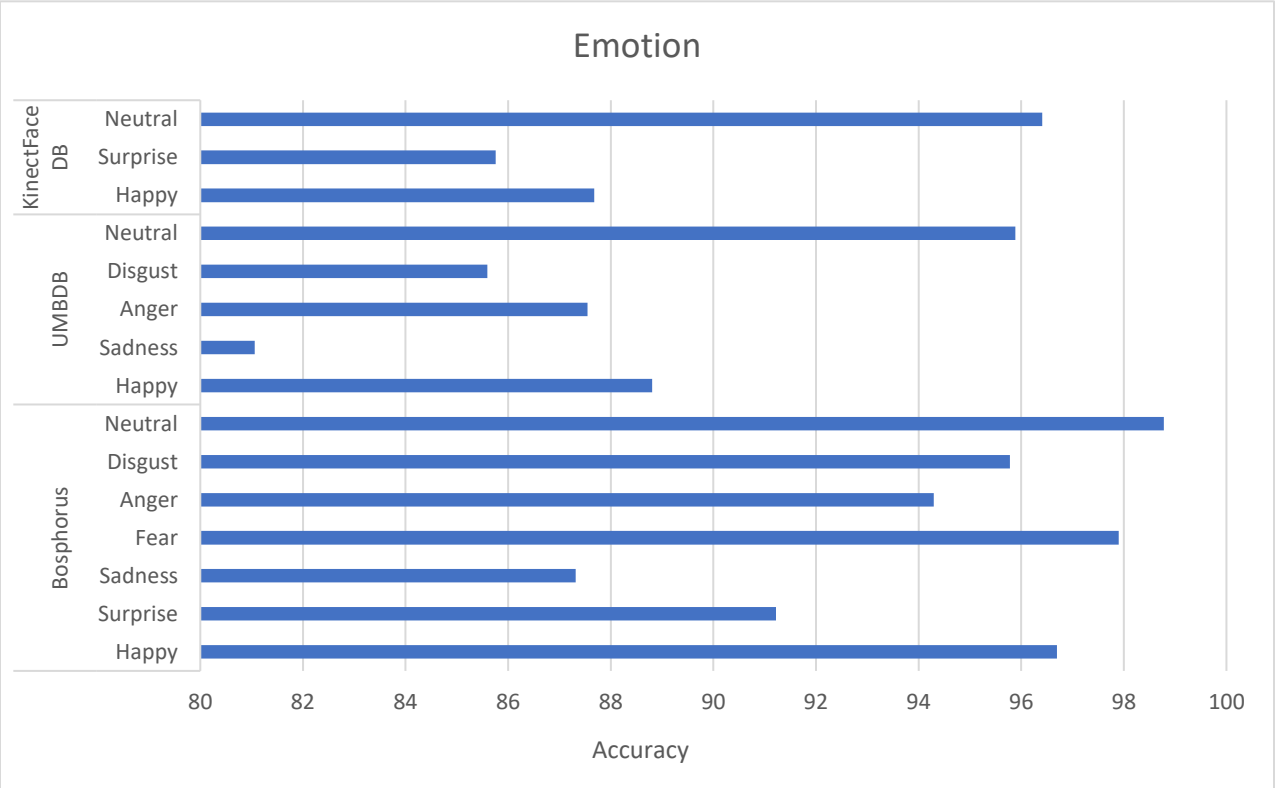


Figure 4.5. Voxel based multiclass accuracy representation for emotion recognition

4.3.3.3 Occlusion Recognition

Table 4.6 shows the different types of occlusions are being classified by the proposed approach. There is 4.07%, 4.82%, and 7.05% increase in accuracy after reconstruction for all three datasets due to the mirror-based technique used in *3DVFR* proposed technique. *3DVFR* uses the mid-plane to generate the voxels missing on one side of the face using the other side voxels of face.

Table 4.6. Quantitative analysis of occlusion recognition using voxels

Metrics **Dataset**	**Accuracy**	**Sensitivity**	**Specificity**	**Precision**	**FPR**	**FNR**	**F1 Score**
	Before 3D Face Reconstruction						
Bosphorus	89.95	95.60	58.74	92.76	41.26	4.40	94.15
UMBDB	76.44	85.12	25.93	86.99	74.07	14.88	86.05
KinectFaceDB	82.80	91.43	29.77	88.89	70.23	8.57	90.14
	After 3D Face Reconstruction						
Bosphorus	94.02	97.31	72.08	95.87	27.91	2.68	96.58
UMBDB	81.26	87.06	50.84	90.27	49.15	12.93	88.64
KinectFaceDB	89.85	93.49	55.55	95.18	44.44	6.50	94.33

Figure 4.6 shows the comparative accuracy of the type of occlusion is represented for each dataset. The occlusion accuracy for Eurecom KinectFaceDB dataset for mouth is 93.3987%, for an eye is 89.5989%, and paper is 86.5523%. For UMBDB dataset, the accuracy for cloth is 78.5853%, hair is 87.1621%, glass is 81.2606%, mouth is 80.2129%, and paper is 79.0788%, respectively. In case of Bosphorus dataset, the accuracy for hair is 98.4778%, glass is 95.5400%, mouth is 88.2121%, and eye is 93.8499%, respectively. In case of face rotation, the reconstruction depends on mirroring technique using mid-face plane. In case of paper, cloth, glass, eye, hair, and mouth, the reconstruction technique builds the missing voxels to some extent in the occluded region. The hair based occlusion is observed as the highest occlusion recognition and cloth is observed as the lowest occlusion recognition.

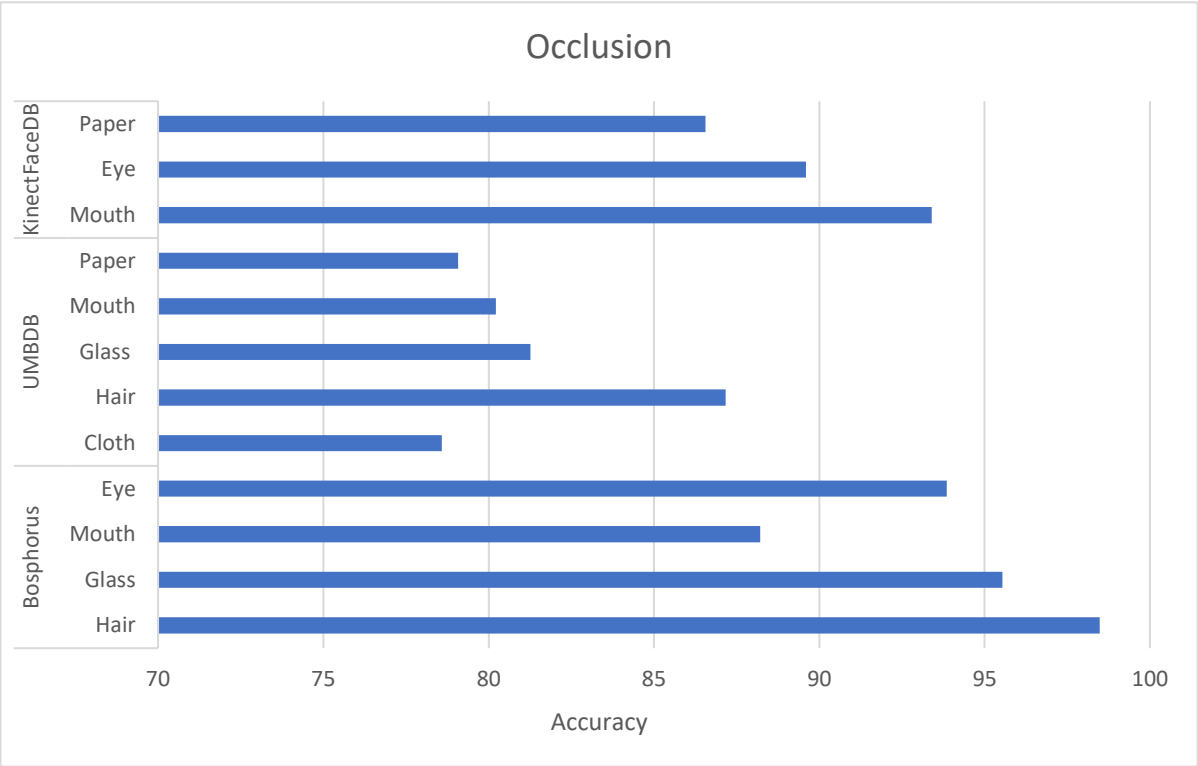


Figure 4.6. Voxel-based multiclass accuracy representation for occlusion recognition

Table 4.7 represents the classification of occlusion types. Bosphorus dataset has four, UMBDB has five and KinectFaceDB has three different types of occlusions in it.

Table 4.7. Occlusion type classification

S.No.	**Dataset**	**Number of occlusions**	**Occlusion type classes**
1.	Bosphorus	4	Hair, Glass, Mouth, Eye.
2.	UMBDB	5	Hair, Glass, Mouth, Cloth, Paper.
3.	KinectFaceDB	3	Mouth, Eye, Paper.

4.3.3.4 Face Recognition

Table 4.8 depicts the quantitative comparison of different 3D face accuracies. The accuracy improvement after reconstruction for three datasets are 9.49%, 8.94%, and 7.45%, respectively.

The number of subjects used are 105, 143, and 52 in Bosphorus, UMBDB, and KinectFaceDB, respectively. In comparison to gender, emotion, and occlusion classes, face recognition has a significantly higher number of classes to be recognized during the training process. The proposed technique has performed considerably well for this number of classes.

Table 4.8. Quantitative results of face recognition using voxels

Metrics **Dataset**	**Accuracy**	**Sensitivity**	**Specificity**	**Precision**	**FPR**	**FNR**	**F1 Score**
	Before 3D Face Reconstruction						
Bosphorus	80.52	87.88	38.12	89.11	61.88	12.12	88.49
UMBDB	69.27	82.13	32.34	82.93	67.66	14.87	84.02
KinectFaceDB	78.23	87.08	33.33	88.62	66.67	7.92	90.31
	After 3D Face Reconstruction						
Bosphorus	90.01	93.38	73.68	94.51	26.32	6.62	93.94
UMBDB	78.21	89.68	31.62	84.19	68.38	10.32	86.85
KinectFaceDB	85.68	92.17	50.00	91.02	50.00	7.83	91.59

4.3.4 Face Recognition Accuracy Based Comparison

The performance of the proposed technique is compared with the different approaches over Bosphorus, UMBDB, and KinectFaceDB datasets after 3D face reconstruction (see Table 4.9). The proposed method *3DVFR* outperforms the existing techniques such as [75], [118], [120]. The best accuracy of face recognition is achieved for Bosphorus dataset. The sensitivity of Feng et al. [120] outperforms *3DVFR* for Bosphorus as well as KinectFaceDB dataset. However, in the case of precision, *3DVFR* outperforms the other approaches. FPR and FNR are low in case of Bosphorus and KinectFaceDB datasets. These results signify the superiority of *3DVFR* in terms of face recognition accuracy.

Table 4.9. Face Recognition Accuracy based comparison of Proposed Approach with other Approaches

Metrics **Dataset**	**Accuracy**	**Sensitivity**	**Specificity**	**Precision**	**FPR**	**FNR**	**F1 Score**
Case 1:	**Bosphorus Dataset**						
Richardson et al. [75]	85.92	91.38	50.95	92.27	49.05	08.62	91.82
Cao et al. [118]	86.67	91.96	41.39	93.07	58.61	08.04	92.51
Feng et al. [120]	87.25	95.56	36.91	90.17	63.09	04.44	92.79
Ours – *3DVFR*	90.01	93.38	73.68	94.51	26.32	06.62	93.94
Case 2:	**UMBDB Dataset**						
Richardson et al. [75]	69.38	77.98	36.60	82.43	63.40	22.02	80.14
Cao et al. [118]	63.41	76.11	25.27	75.36	74.73	23.89	75.73
Feng et al. [120]	73.12	87.14	38.30	77.81	61.70	12.86	82.21
Ours – *3DVFR*	78.21	89.68	31.62	84.19	68.63	10.32	86.85
Case 3:	**KinectFaceDB Dataset**						
Richardson et al. [75]	76.50	84.19	47.45	85.81	52.55	15.81	84.99
Cao et al. [118]	67.20	77.96	30.00	79.38	70.00	22.04	78.67
Feng et al. [120]	81.73	93.17	36.51	85.29	63.49	06.83	89.06
Ours – *3DVFR*	85.68	92.17	50.00	91.02	50.00	07.83	91.59

4.3.5 Computational Time Analysis

Table 4.10 shows the time taken in seconds by a single probe image for three steps in the pipeline of architecture, pre-processing, reconstruction, and verification. In case of pre-processing, the proposed approach (*3DVFR)* outperform the other techniques [75], [118], [120] by completing the pre-processing phase in 0.078 seconds. In addition, it outperforms the other techniques in reconstruction phase by completing the task in 0.060 seconds as the mean time. *3DVFR* outperforms the other techniques during verification phase with an average time of 0.098 seconds for all datasets. *3DVFR* has faster computation on a GPU than the other methods.

Table 4.10. Comparison of computation time (seconds) of reconstruction per probe (using GPU) and deep learning technique

Approaches	**Pre-processing**	**Reconstruction**	**Verification**	**Deep learning model**
Richardson et al. [75]	0.087	0.071	0.120	CNN
Cao et al. [118]	0.082	0.090	0.117	CNN
Feng et al. [120]	0.103	0.065	0.106	CNN
3DVFR	**0.078**	**0.060**	**0.098**	VAE, BiLSTM

4.3.6 Convergence Analysis

The convergence of *VAE* is shown in Figure 4.7. For 200 voxels and 1200 epochs, VAE converges at accuracy of 70%. For 400 voxels and 1200 epochs, VAE converges at accuracy of 73%. Similarly, for 800 voxels and 1100 epochs, convergence is found at accuracy of 75%. The accuracy of VAE can be increased by using BiLSTM based on its ability to learn the long term dependencies using forward and backward propagation.

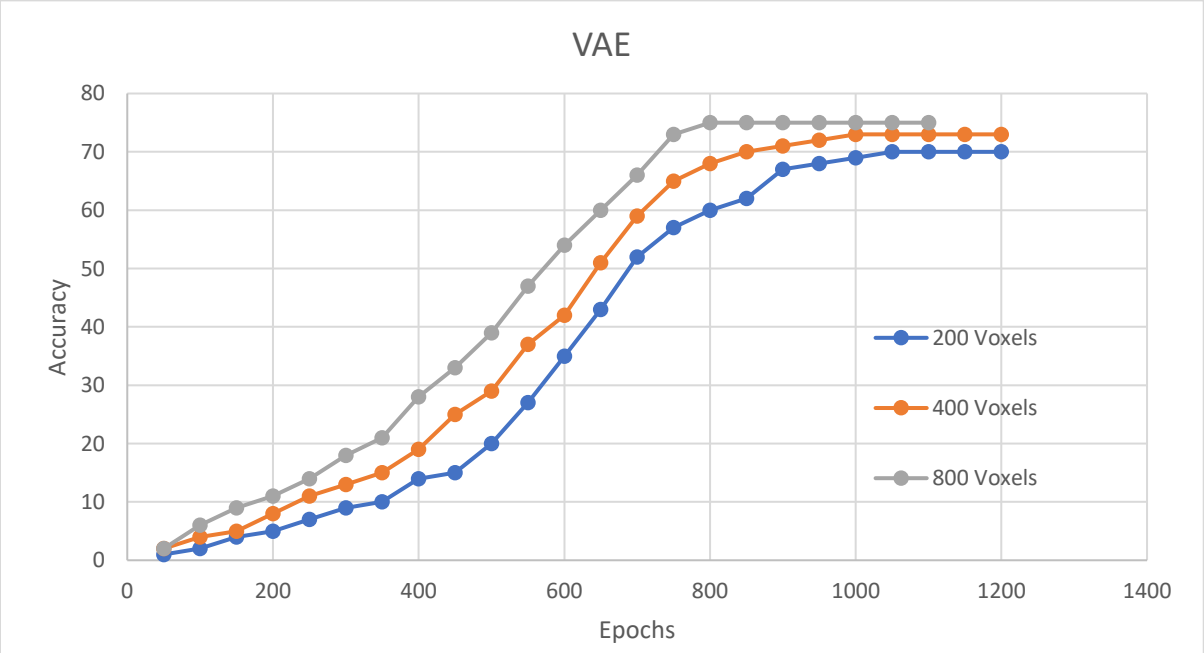


Figure 4.7. Convergence of voxel based accuracy for VAE

The convergence of *BiLSTM* is shown in Figure 4.8. For 200 voxels and 240 epochs, convergence is at accuracy of 80%. For 400 voxels and 230 epochs, convergence is at accuracy of 84%. Similarly, for 800 voxels and 220 epochs, convergence is at accuracy of 86%. This accuracy can be optimized using triplet loss technique, which is capable of handling the imbalanced classes in datasets.

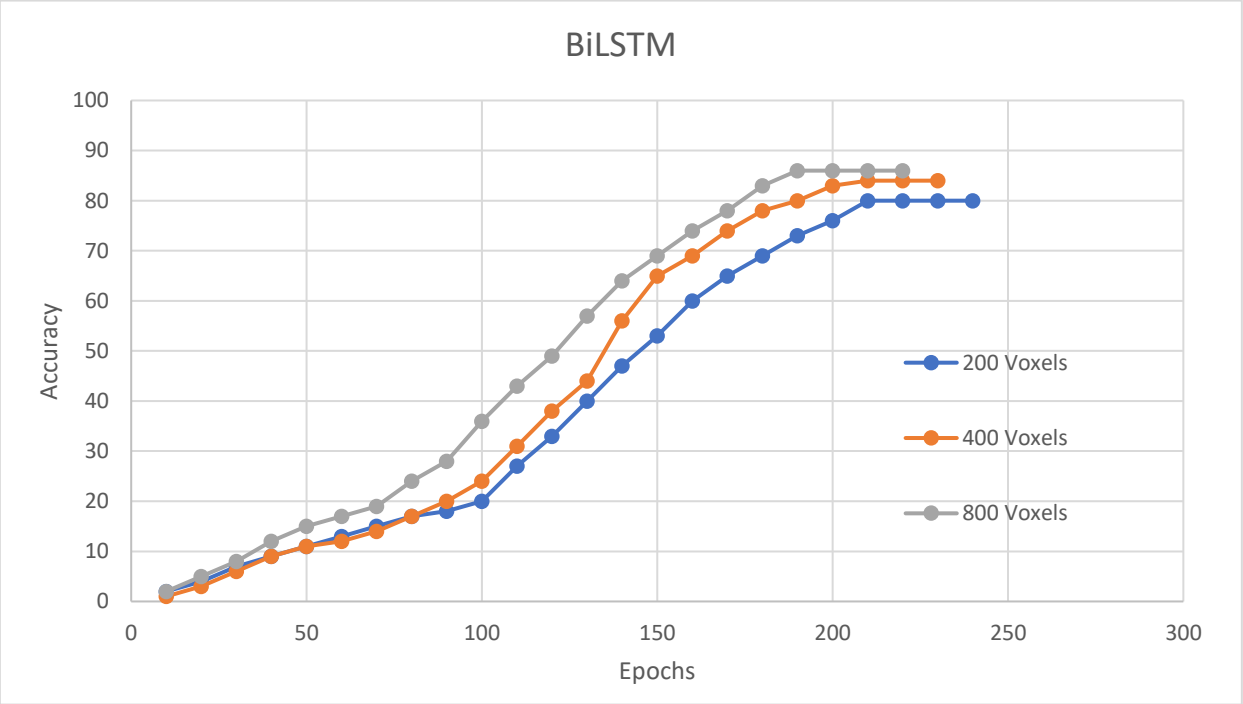


Figure 4.8. Convergence of voxel based accuracy for BiLSTM

The convergence of *triplet loss with SVM* is shown in Figure 4.9. For 200 voxels and 3050 epochs, convergence is at 15% error. For 400 voxels and 2850 epochs, convergence is at 8% error. Similarly, for 800 voxels and 2700 epochs, convergence is at 7% error rate. Hence, the usage of pipeline in deep learning has helped in the improvement of final accuracy of the proposed approach.

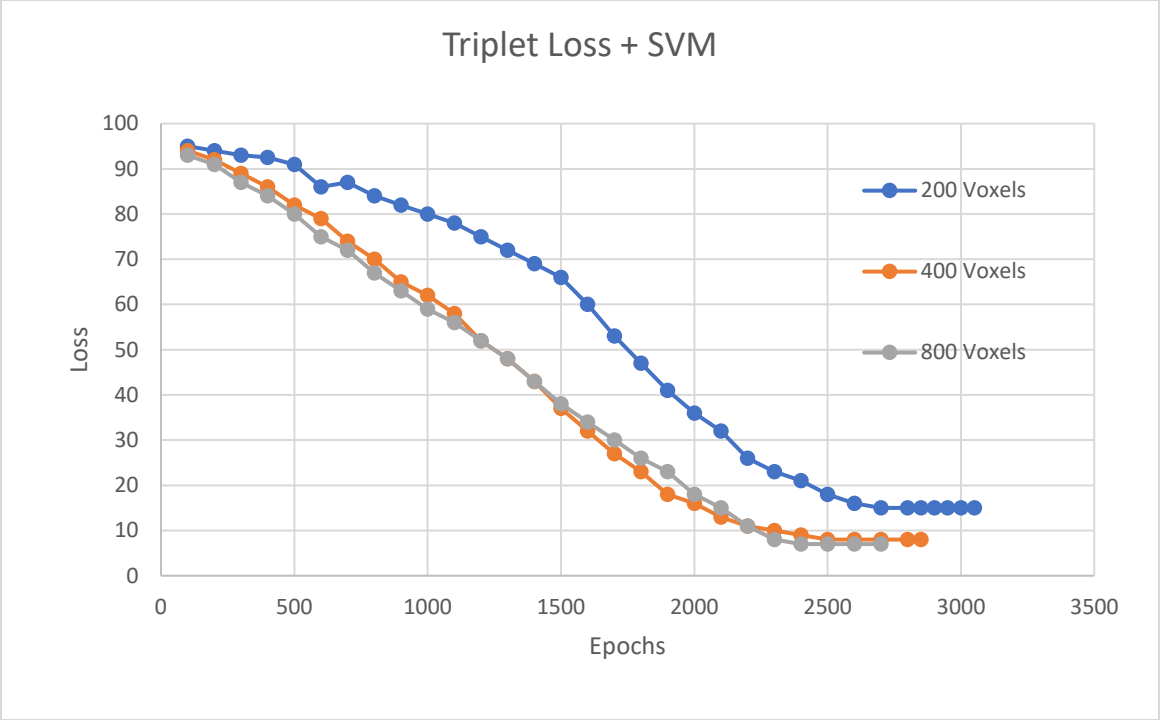


Figure 4.9. Cumulative loss plot of Triplet loss and SVM

4.4 Summary

In this chapter, a novel 3D face recognition framework is proposed, which is based on deep sequential learning. 3D voxel-based face reconstruction (*3DVFR*) is motivated from the concept of mirror image. The proposed deep learning pipeline utilizes the concept of variational autoencoder (*VAE*), bidirectional long short-term memory (*BiLSTM*), and triplet loss training. FCM technique helps in handling the sparsity of voxels obtained after voxelization technique for different poses. *VAE* helps in an efficient dimensionality reduction of the number of voxels by using the probabilistic model for latent space. *BiLSTM* technique uses the sequential data property of facial features to optimize *VAE* embedding. Triplet loss training technique helps in reducing the loss. A support vector machine-based classifier is used to predict the gender, emotion, occlusion, and person classification. Experiments have been performed on three publicly available 3D face datasets, namely Bosphorus, UMBDB, and KinectFaceDB, respectively. In performance evaluation phase, all the predictions are demonstrated with and without the 3D face reconstruction. In gender recognition, the accuracy is above 90% after reconstruction for all the three datasets due to binary classification. In emotion recognition, Bosphorus dataset accuracy is 94.57% after

reconstruction as compared to 83.95% before the reconstruction, i.e., 10.62% increase in accuracy. In case of UMBDB and KinectFaceDB datasets, there is an increase of 14.9% and 12.49% accuracies, respectively. In occlusion recognition, there is an increase of 4.07%, 4.82%, and 7.05% for Bosphorus, UMBDB, and KinectFaceDB datasets, respectively. The accuracy of the proposed framework (*3DVFR*) is state-of-the-art with 90.01% in 3D face recognition using voxels for Bosphorus dataset. In case of UMBDB and KinectFaceDB datasets, the face recognition accuracies are 78.21% and 85.68%, respectively. *3DVFR* has been compared for accuracy as well as the computational time with the other methods. 3DVFR outperforms the other methods by 3-5% increase in accuracy in case of all the datasets.

5. 3D Landmark-based Face Restoration Using Variational Autoencoder and Triplet Loss

5.1 Introduction

Restoration of a 3D face from the mesh image is highly demanded in computer vision applications. 3D face restoration is a challenging task due to the variation of expression, poses, intrinsic geometries, and textures. This chapter presents a proposed technique consisting of two main components, namely face restoration and recognition. A novel three-dimensional (3D) landmark-based face restoration method is proposed. 3D facial landmarks are used in the face recognition technique. It uses the principle of reflection and mid-face plane for the restoration of facial landmarks. By using the restored 3D face, a deep learning-based face recognition system is developed. It utilizes the concept of deep features from variational autoencoders. Further, these deep feature embeddings are trained using triplet loss training to increase the distance between embeddings of different persons and decrease the distance between embeddings of the same person. These trained embeddings are used in support vector machine for prediction. The proposed framework is compared with recently developed face recognition techniques in terms of computational time. The proposed technique is able to recognize the person's face with better accuracy than the existing methods. Further, ablation studies are conducted to test the robustness of the proposed technique.

5.2 Proposed 3D Face Recognition Framework

This section discusses the motivation behind the face recognition technique followed by restoration-based 3D face recognition framework.

5.2.1 Motivation

The concept of facial asymmetry has been studied in the field of plastic surgeries and reconstruction. Facial symmetry is one of the key factors in human attractiveness [202]–[205]. Linden et al. [206] presented the study of age effect on facial symmetry. These facts motivated me

for developing 3D face recognition framework. From a holistic point of view, more than 80% of the face is symmetrical in nature. Sharma and Kumar presented 3D voxel-based face reconstruction (3DVFR), which is present in Chapter 4. It uses full mesh as input and generates reconstructed voxels. It performs voxelization process for reconstructing the reconstructable voxels. The computational time is more for voxel reconstruction as compared to landmark reconstruction. It gives us a new direction to develop the computationally efficient face recognition technique.

5.2.2 Proposed 3D Face Recognition Framework

The proposed 3D deep learning based face recognition framework is shown in Figure 5.1. It is based on deep learning and restoration of 3D landmarks. While designing the pipeline, once ground truth landmarks are fetched from a 3D mesh image. The number of landmarks is sparse for different images due to the variation of pose. The proposed restoration technique restores the missing landmarks using mirroring technique. The raw 3D landmarks are to be refined using deep learning for prediction phase. VAE uses the concept of probabilistic encoder to remove the sparsity of 3D landmarks. The deep features obtained from VAE are optimized using triplet loss training method. There are two phases namely, training and testing. The detailed description of these phases is as follows.

5.2.2.1 Training Phase

The training phase of the proposed framework includes pre-processing of 3D mesh image, the restoration process, deep learning-based models, and prediction of face after restoration from occlusion.

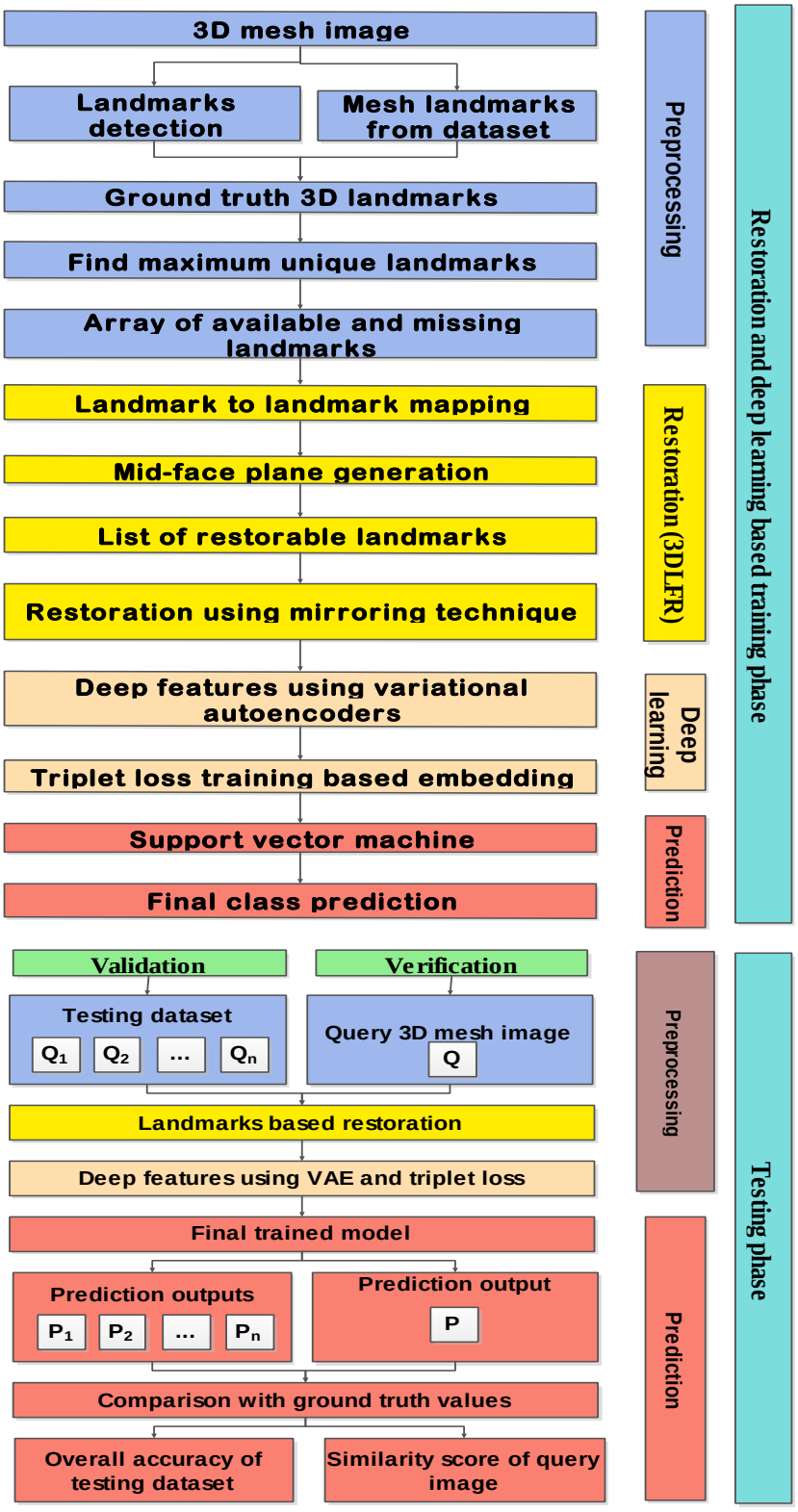
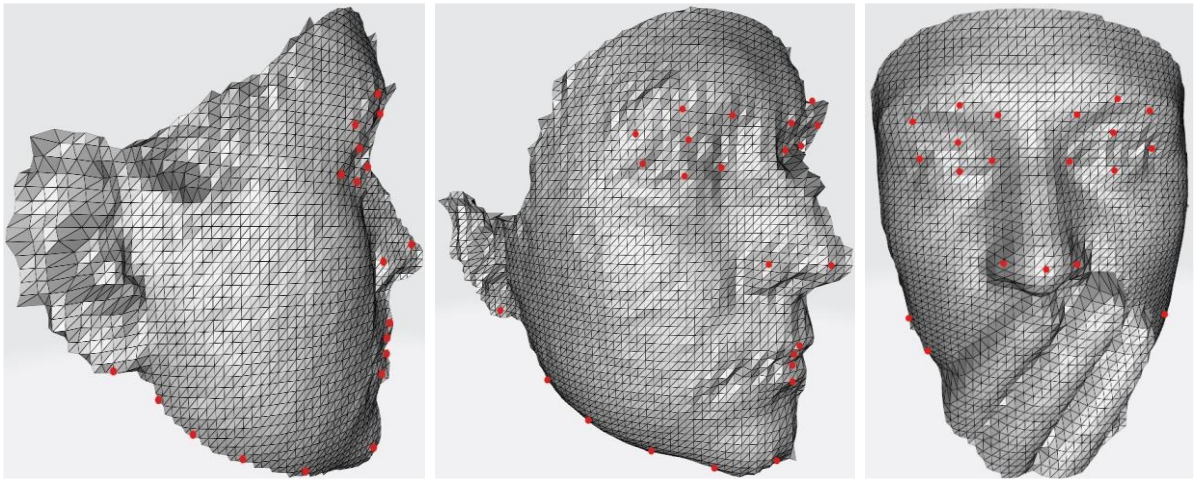


Figure 5.1. Proposed 3D Deep Learning based Face Recognition Framework

A. Pre-Processing

3D facial landmarks are extracted from the mesh image. These landmarks may or may not be present within the dataset. While extracting the face landmarks, the data augmentation technique plays a vital role in improving the quality of facial landmarks. The extracted landmarks serve as ground truth landmarks. Unique landmarks are generated for all the images in the dataset. The landmarks of each image are compared to a list of unique landmarks for generating the array of available and missing landmarks using landmark-to-landmark mapping. The number of landmarks varies for all images. As depicted in Figure 5.2, a face with yaw rotation left 90° has 19 landmarks initially. Whereas, a face with yaw rotation of left 45° has 24 landmarks.



a. Pose: Left 90°
Landmarks: 19

b. Pose: Left 45°
Landmarks: 24

c. Pose: Occlusion
Landmarks: 20

Figure 5.2. Representation of 3D landmarks under different pose meshes

B. Proposed Restoration Algorithm and Computational Complexity

The restoration phase of the proposed framework utilizes the concept of 3D landmarks-based face restoration (*3DLFR*). Training (T_r) and testing (T_s) sets are randomly selected from the given dataset. Mesh image (I_m) is taken as input from T_r. Initially, the array of ground truth landmarks (L_{GT}) is fetched in correspondence to I_m. Thereafter, an array L_{Mid} of mid-face landmarks is generated (*Step 2*). The selection of one of the landmarks (L_M), and nose tip landmark (L_{NT}) is done

from array L_{Mid} (*Step 3*). The next step is to generate one landmark (L) such that it lies on the same mid-face plane as of L_{NT} (*Steps 4 and 5*). In *Step 6*, the midface plane is generated using L_{NT}, L, and L_M. A list of reconstructable landmarks is calculated by the technique of $\{L_{GT} - L_{Mid}\}$ (*Step 7*). Finally, the reconstruction is achieved by mirroring technique (*Step 8*). Restoration using a mirroring technique through a mid-face plane is implemented from the list of reconstructable landmarks. Algorithm 5.1 depicts the 3D landmark-based face restoration technique.

Algorithm 5.1. *3DLFR – 3D landmarks-based face restoration technique*

Input: *Dataset D is divided into Train (T_r) and Test (T_s) sets randomly. Mesh image I_m taken as input from the T_r.*

Output: *Restored 3D landmarks*

1. Fetch array of ground truth landmarks (L_{GT}) corresponding to I_m.
   a. $L_{GT} = [L_1, L_2, L_3, \ldots, L_n]$
2. Generate an array L_{Mid} of mid-face landmarks from array L_{GT} which are on mid-face plane.
   a. $L_{Mid} = [L_{NT}, L_{ULUM}, L_{ULLM}, L_{LLUM}, L_{LLLM}, L_{CM}]$
3. Select nose tip face landmark L_{NT} from L_{Mid} and one more landmark L_M from L_{Mid} as mid-face landmark from L_{GT}.
   a. $L_{NT} = (x_{00},y_{00},z_{00})$
   b. $L_M = (x_{11},y_{11},z_{11})$
4. Find maximum and minimum z coordinate (z_{min} and z_{max}) from L_{GT}.
5. Choose one random $L(x_{00},y_{00},z)$ such that $z_{min} < z < z_{max}$.
6. Using $L_{NT}(x_{00},y_{00},z_{00})$, $L(x_{00},y_{00},z)$, and $L_M(x_{11},y_{11},z_{11})$, calculate 3D mid-face plane.
7. Find the array (R) of all restorable points on face.
   $R = \{L_{GT} - L_{Mid}\}$ //List of landmarks not on mid-face
8. Restore the 3D face landmarks using the concept of mirroring through mid-face plane for all the R number of points on the face.
   l. *For t = 1 to len(R) do*
   m. $P = R[t] = [x_1, y_1, z_1]$ *//P is restorable 3D point*
   n. $N = [x_2, y_2, z_2]$ *//PN is perpendicular to mid-face plane*
   o. *PN* direction ratios = *[a, b, c] //a, b, c are constants*
   p. $PN = [(x-x_1)/a = (y-y_1)/b = (z-z_1)/c = k]$ *//k is constant*
   q. $x = ak + x_1, y = bk + y_1, z = ck + z_1$
   r. Since *N* lies on mid-face plane
   s. $k=(-ax_1-by_1-cz_1-d)/(a^2+b^2+c^2)$
   t. $x_2=ak+x_1, y_2=bk+y_1, z_2=ck+z_1$
   u. return $[2x_2-x_1, 2y_2-y_1, 2z_2-z_1]$ *//Restored 3D landmark*
   v. *End for*

The face restoration phase aims to generate the missing landmarks by separating these landmarks either on the mid-face plane (L_{Mid}) or not on the mid-face plane (R). By using the mid-face plane, the restoration of the landmarks is done in *Step 8*. Restoring the missing landmarks helps in increasing the deep learning accuracy by providing suitable parameter setting during the model training.

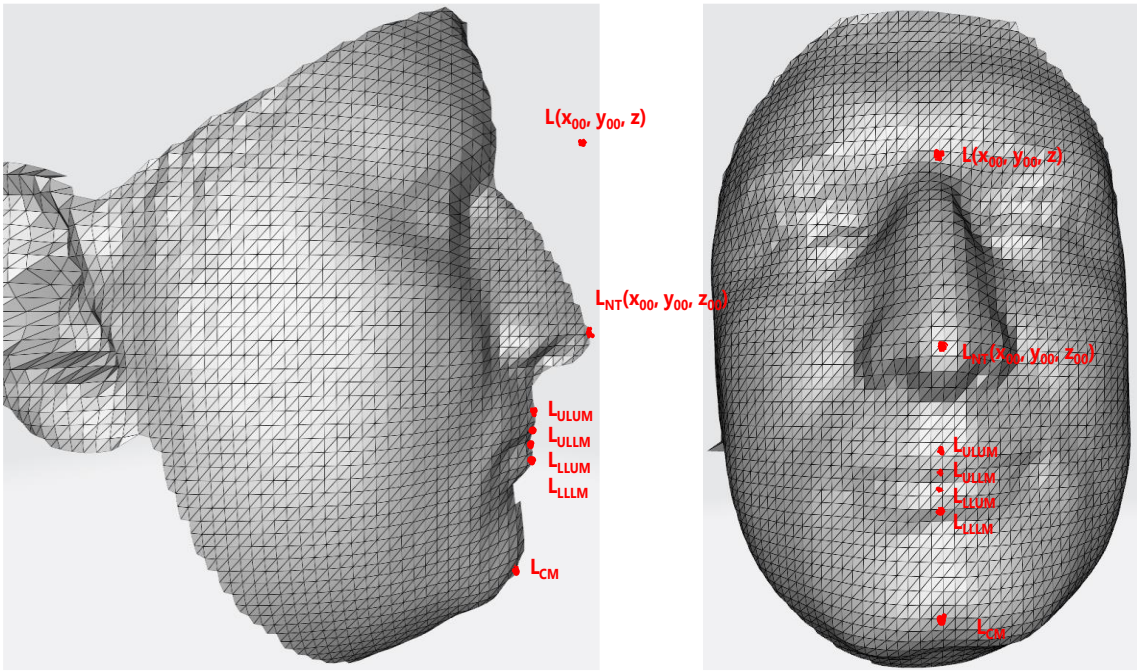


Figure 5.3 Selection of three points on face to calculate mid-face plane

Figure 5.3 illustrates how the three points are selected for calculating the mid-face plane. The first point is the nose tip $L_{NT}(x_{00},y_{00},z_{00})$ which is assumed to be available in all 3D face meshes. The second landmark needed for mid-face plane generation is one of the mid-face landmark $L_M(x_{11},y_{11},z_{11})$. It can be chin middle (L_{CM}), upper lip upper-middle (L_{ULUM}), upper lip lower middle (L_{ULLM}), lower lip upper-middle (L_{LLUM}), lower lip lower middle (L_{LLLM}), mid-point of the two eyebrows, and mid-point of the two eyes landmarks. The third point is calculated by varying the value of z in the range of z_{min} and z_{max}, by keeping (x_{00},y_{00}) as it is. Using this technique, the three points are not collinear and a unique midface plane can be easily formed for mirroring of reconstructable landmarks (R). Hence, the total number (T) of landmarks after reconstruction becomes

$$T = (2 \times len(R) + len(L_{Mid})) \quad (5.1)$$

This method increases the number of landmarks to double. The number of landmarks is given as input to VAE model for generating deep features. Figure 5.4 depicts 3D landmarks before and after the restoration by mirroring.

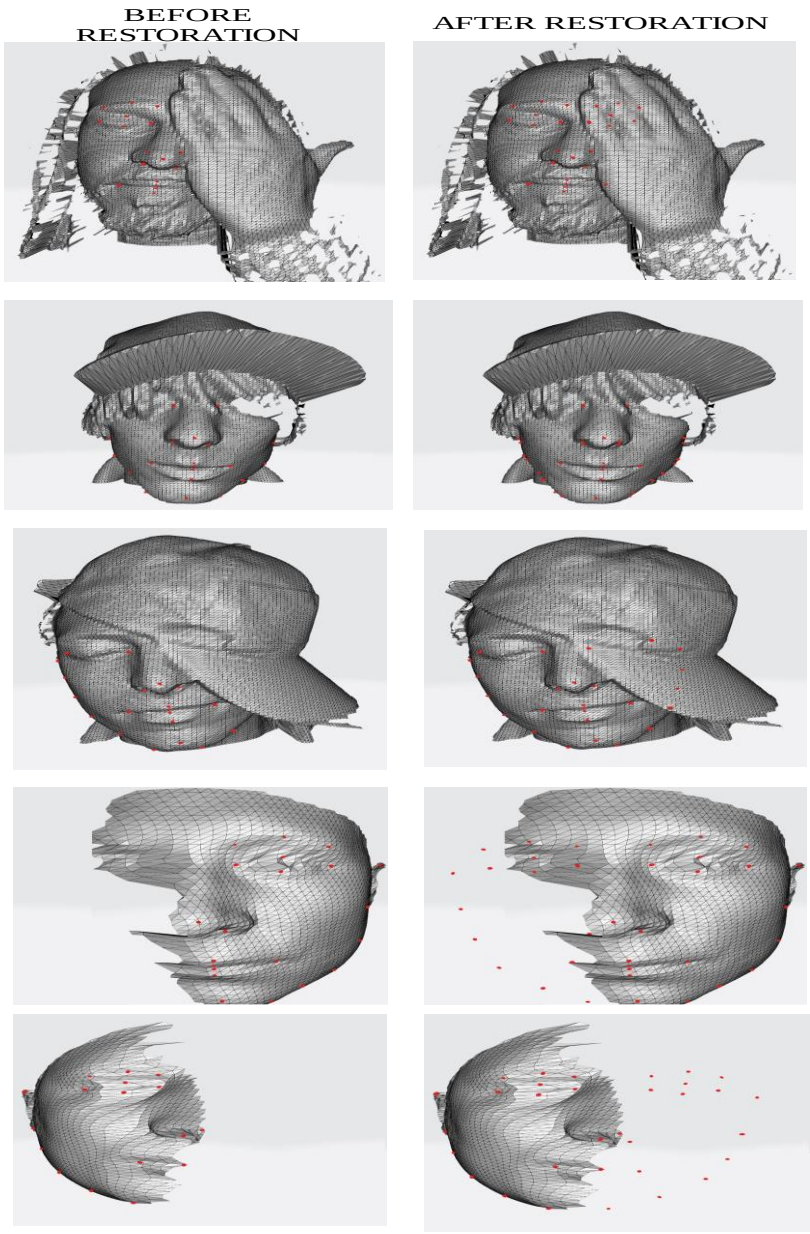


Figure 5.4. Visualization of Landmarks Before and After Restoration

The time complexity of the proposed algorithm is as follows. The fetching of ground truth landmarks (Step 1) requires *O(1)* time for a single mesh. The generation of the array L_{Mid} (Step 2) requires *O(L)* time. In Step 3, selecting L_{NT} and L_M requires *O(1)* time, and finding the maximum-

minimum z coordinates (Step 4) requires *O (L)* time each. Choosing one random value (Step 5) between z_{min} and z_{max} requires *O(1)* time. In Step 6 and 7, finding the mid-face plane as well as the list of restorable points on the face requires *O(1)* time. In the final Step, the restoration process needs *O(R)* time. Hence, the asymptotic time complexity of the proposed technique is *O(L)*, where *L* is the number of facial landmarks.

The limitation of the proposed algorithm is that it does not work when there is no symmetry left. The face is said to be with no symmetry when the nose tip is missing, or the face is hidden for more than 50% of the area. Preferably, the nose tip should always be visible for proceeding in the restoration.

C. Deep Learning Phase

After the restoration of the landmarks, the deep learning phase is implemented. The total number of landmarks (T) are given as input to variational autoencoders (VAE) for extraction of deep features. Variational autoencoders are used for enhancement of the restored landmarks due to probabilistic encoding and decoding capability. Original 3D landmarks, as well as restored landmarks, are both combined for this phase. The main advantage of using VAE after the restoration is that it removes the sparsity of landmarks. Further, triplet loss training is performed on deep features for fine-tuning of feature embedding in the prediction phase.

D. Prediction Phase

The success of support vector machine (SVM) in classification [195], [198], [207], [208] made us to use it for predicting the subject identification number. 3D face landmarks after restoration are treated as a medium of coarse-to-fine features. Hence, the chosen variant of SVM is medium gaussian SVM. The details of the parameter setting of medium Gaussian SVM are mentioned in Section 5.3.2.

5.2.2.2 Testing Phase

In testing phase, the concepts of validation and verification are represented in pre-processing and prediction sub-phases.

A. Pre-Processing

The validation is performed on the testing dataset. The ground truth landmarks are used for

restoration. Enhanced embedding of the restored landmarks is done by using VAE and triplet loss training. The embeddings generated through deep learning are given as input to the final trained model of the training phase.

B. Prediction

All the predictions are recorded and compared with ground truth values. Overall accuracy is calculated from the correctness of predictions. For verification purposes, a single query image is processed for prediction and compared with ground truth value to generate a similarity score.

5.3 Results and Discussion

This section describes the details of datasets used, parameter setting, performance evaluation and metrics, computational time analysis of the proposed techniques, and their convergence.

5.3.1 Parameter Setting

The parameters of the proposed technique and compared techniques are depicted in Table 5.1. The parameter setting of the proposed technique is set by trial and error method while keeping variational autoencoder [54], triplet loss training[209], and support vector machine [207], [208]. In variational autoencoders, the number of epochs and batch size are set to 1200, and 50, respectively. The activation function is set as rectified linear unit (*ReLU*). L2 regularization is employed along with adaptive moment (*Adam*) optimizer. In triplet loss training, the number of epochs is set to 2300 with alpha value as 0.2 using *Adam* optimizer. Mean absolute error (MAE) is used for convergence computation. During the prediction phase, medium gaussian support vector machine is used with Gaussian kernel function and kernel scale is set to 9.1. The data augmentation based pre-processing technique was added for improving 3D face landmarks. There are three parameters such as width-shift range, height-shift range, and zoom range. All of these parameters are set to 0.2.

Table 5.1. Parameter setting for the involved algorithms

Proposed Technique/Parameter	**Value**
Variational Autoencoders	
Number of Epochs	1200
Batch Size	50
Activation	ReLU
Regularization	L2
Latent Dimension	200
Optimizer	Adam
Triplet loss training	
Number of epochs	2300
Optimizer	Adam
Alpha	0.2
Loss	Mean Absolute Error
Support Vector Machine	
Type	Medium Gaussian SVM
Kernel Function	Gaussian
Kernel Scale	9.1
Multiclass Method	One vs Rest
Richardson et al. [75]	
Input size	200x200
CNN model	ResNet, VGG
Loss	Mean Square Error
Objective function	Shape from Shading
Hsu et al. [117]	
Input size	128 x 128
CNN model	Revised VGG
Optimizer	TCDCN [51]
Activation	ReLU
Dropout	0.5
Gecer et al. [39]	
GAN Type	Progressive Growing GAN
Image Size	512 x 512
Optimizer	Adam
Identity Loss	Cosine Distance
Content Loss	Normalized Euclidean Distance

The parameters for state-of-the-art algorithms are set as reported in the literature [39], [75], [117]. Richardson et al. [75] have an image 200x200 on ResNet [112], and VGGNet [210]. Mean square error is used for calculating loss value and shape from shading as the objective function. Hsu et al. [117] utilized VGGNet [210] with input image size of 128x128. Gecer et al. [39] used progressive growing GAN [211] with an input image size of 512x512, normalized Euclidean distance for content loss computation, and *Adam* optimizer.

5.3.2 Performance Evaluation

Table 5.2 depicts the accuracy, sensitivity, and specificity obtained from face recognition techniques before and after the restoration. There is 2-8% increase in accuracy of face recognition after restoration in contrast to before restoration process for Bosphorus, UMBDB, and KinectFaceDB datasets. Moreover, the proposed method performs better than the other approaches in terms of 3-8% more accuracy for all the datasets. The reason behind the better performance of the proposed technique is due to clustering-based triplet loss will decrease the distance between the embeddings of the same class and increase the comparative distance of embeddings of a different class.

Table 5.2. Performance evaluation of face recognition techniques

Datasets	**Accuracy**				**Sensitivity**				**Specificity**			
	Proposed	[75]	[117]	[39]	Proposed	[75]	[117]	[39]	Proposed	[75]	[117]	[39]
	Before Restoration											
Bosphorus	89.6	78.2	83.5	85.2	94.9	85.5	92.8	93.1	58.4.	29.9	19.7	56.1
UMBDB	79.8	70.9	72.8	74.9	89.5	81.2	81.3	84.9	36.9	29	35.5	26.3
Kinect FaceDB	88.0	75.4	80.4	79.2	93.4	86.4	88.8	87.3	63.3	14.7	29	38
	After Restoration											
Bosphorus	93.5	85.92	88	90.5	97.2	91.6	93.5	94.4	62.4	39.3	45.8	50.7
UMBDB	87.9	78.68	83.5	82.9	95.33	85.8	92.7	89.9	53.3	37.5	36	48.6
Kinect FaceDB	92.0	84.29	85.9	83.7	95.2	90.3	92.8	89.8	77.7	47.7	39.7	51.7

The experimentation for comparing the performance of face recognition is divided into two parts for all methods, before restoration and after restoration. Here, before restoration means that the performance metrics such as accuracy, sensitivity, and specificity are calculated without the restoration of landmarks for the proposed technique. The performance of proposed technique before restoration is compared with Gecer et al. [39], Hsu et al. [117], and Richardson et al. [75] techniques. The performance of these methods was calculated by removing texture GAN from [39]. It was done by removing 3D reconstruction phase of [117] from the training phase, and by removing the FineNet approach from [75]. It use only CoarseNet for prediction of face recognition. The whole idea of this technique is to compare the state-of-the-art techniques with the proposed method before and after the reconstruction.

The accuracy, sensitivity, and specificity measures in all the cases are best for the proposed technique. The obtained accuracy for UMBDB dataset is below 90% as compared to Bosphorus and KinectFaceDb datasets. The major challenge for UMBDB dataset is that it contains few occlusions are upto 80% face occlusion, including the missing of nose tip. Some faces have only eyes visible, head, mouth, chin, and nose covered with cap and cloth.

The proposed technique has performed best for Bosphorus dataset, the second-best for KinectFace dataset, and lowest for UMBDB dataset. The reason for lowest accuracy for UMBDB is that it has the highest classes of occlusions available in the dataset as compared to the other two datasets.

5.3.3 Computational Time Analysis

Table 5.3 shows the computational time analysis of the proposed techniques with other techniques. All the computation is done using Nvidia GTX 1080Ti GPU. The comparison is made for three phases viz. pre-processing, restoration, and verification.

Table 5.3. Computational time analysis versus other approaches (per probe)

Approach	Pre-processing (ms)	Restoration (ms)	Verification (ms)	Deep learning model
Richardson et al. [75]	87	71	120	CNN
Hsu et al. [117]	82	90	117	CNN
Gecer et al. [39]	80	78	110	GAN
Proposed Approach	55	52	80	VAE

The computational time for the proposed technique is 55 ms for pre-processing, 52 ms for restoration, and 80 ms for verification of the single probe image using VAE deep learning model, which is 50% faster than the existing approaches.

5.3.4 Convergence Analysis

The convergence of the proposed technique is used to demonstrate the consistency of trained model. Figure 5.5 describes the convergence of accuracy for VAE and triplet loss with SVM. VAE converges to 75% accuracy in 1100 epochs and is terminated at 1200 epochs. Triplet loss with SVM converges at 93.5% accuracy in 2200 epochs and is terminated at 2300 epochs. It is clear that VAE converges faster as compared to triplet loss and SVM training. The training is stopped using the callback function.

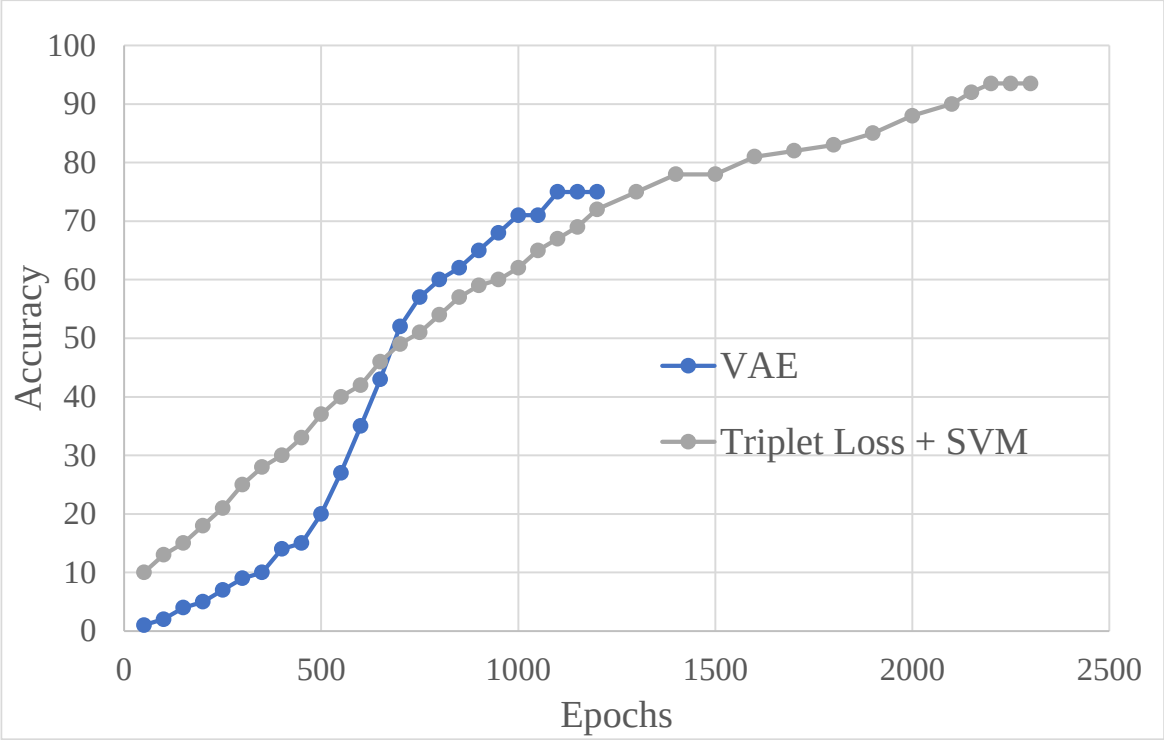


Figure 5.5. Convergence of accuracy for VAE and triplet loss with SVM

5.4 Ablation Studies

This section presents four experimentations to support the ablation studies. The different experimentations include occluded versus non-occluded faces, gender recognition, emotion class recognition, and the type of occlusion recognition.

5.4.1 Occluded versus Non-Occluded Faces

The performance of the proposed face recognition framework is presented in this subsection. All three datasets are tested for performance metrics viz. accuracy, sensitivity, and specificity using occluded as well as non-occluded faces. Table 5.4 presents the results for facial landmarks with and without restoration. For UMBDB, the accuracy is more than 10% in case of reconstruction of occluded face. There is a consistent increase in performance accuracy for both instances of the occluded face and non-occluded faces for all datasets. Additionally, the sensitivity and specificity are improving after restoration for all the cases. For Bosphorus dataset, the best accuracy for

occluded faces is 92.60%, and for non-occluded faces is 93.83%.

Table 5.4. Performance of proposed technique for occluded versus non-occluded faces

Dataset	**Restoration**	**Occluded Face**			**Non-Occluded Faces**		
		Accuracy	**Sensitivity**	**Specificity**	**Accuracy**	**Sensitivity**	**Specificity**
Bosphorus	×	89.54	93.53	61.87	90.17	95.00	62.27
	✓	92.60	95.65	70.74	93.83	96.68	77.80
UMBDB	×	76.31	89.03	37.24	83.50	90.24	48.64
	✓	87.98	93.57	55.93	87.71	92.32	56.63
KinectFaceDB	×	83.97	92.26	54.10	89.32	95.80	58.67
	✓	89.90	92.43	66.20	89.98	96.32	59.38

5.4.2 Gender Recognition

The performance of the proposed face recognition framework is presented for gender classification. Table 5.5 shows the performance of the proposed technique on gender recognition.

Table 5.5. Performance on Gender Recognition

Metrics	**Accuracy**	**Sensitivity**	**Specificity**
Dataset	Before 3D Face Restoration		
Bosphorus	81.0	91.2	68.6
UMBDB	69.5	85.6	48.2
KinectFaceDB	77.4	89.4	65.5
	After 3D Face Restoration		
Bosphorus	89.8	92.6	78.4
UMBDB	80.7	86.9	70.3
KinectFaceDB	85.6	86.1	73.7

It can be seen from Table 5.6 that the increase in accuracy, sensitivity, and specificity is consistent for all three datasets after 3D face restoration. There is 8-11% increase in gender recognition accuracy for all three datasets. The best accuracy for gender recognition after restoration is 89.8% for Bosphorus dataset. There is 1-3% increase in sensitivity and 20% increase in specificity after 3D face restoration for UMBDB dataset.

5.4.3 Emotion Recognition

This sub-section deals with prediction of the proposed framework for emotion recognition. Table 5.6 presents the different classification of emotions for all three datasets.

Table 5.6. Classification of Emotions

Dataset	#Classes	Types
Bosphorus	7	Anger, Disgust, Fear, Happy, Neutral, Sadness, Surprise
UMBDB	5	Anger, Disgust, Happy, Neutral, Sadness
KinectFaceDB	3	Happy, Neutral, Surprise

Table 5.7 presents the performance of the proposed technique for prediction of emotion recognition. Notably, the accuracy, sensitivity, and specificity are improved after 3D face restoration. There is a 10% increase in emotion recognition for Bosphorus dataset to reach 93.5% after face restoration. In terms of sensitivity, the values achieved are an increase of 3% after 3D face restoration. For specificity, the rise is between 11-14% after 3D face restoration.

Table 5.7. Performance on Emotion Recognition

Metrics	Accuracy	Sensitivity	Specificity
Dataset	Before 3D Face Restoration		
Bosphorus	83.3	91.1	65.8
UMBDB	77.2	84.7	57.3
KinectFaceDB	80.9	86.9	61.4
	After 3D Face Restoration		
Bosphorus	93.5	94.5	79.2
UMBDB	84.2	87.1	69.1
KinectFaceDB	87.8	91.6	72.7

5.4.4 Occlusion Recognition

There are two types of occlusions namely, Internal Occlusion, and External Occlusion. Internal occlusions are those occlusions, which are due to the person's own body parts or posing. External occlusions are those occlusions, which are not body part. It might be some object or other person blocking the view. Table 5.8 presents all the internal and external occlusions available in the three datasets.

Table 5.8. Classification of Internal and External Occlusion

Dataset	Internal Occlusion	External Occlusion
Bosphorus	Hair, Mouth, Eye, Yaw Rotation	Glass
UMBDB	Hair, Glass, Mouth, Yaw Rotation	Cloth, Paper, Cigarette, Bottle
KinectFaceDB	Mouth, Eye, Yaw Rotation	Paper

Table 5.9 presents the performance metrics for internal and external occlusion recognition. Notably, the accuracy, sensitivity, and specificity are improved after 3D face restoration for all

three datasets. The highest accuracy is achieved for Bosphorus dataset with 94.2%, and the lowest is 85.8% for UMBDB dataset. A rise of 3-8% is seen in sensitivity after the restoration for occlusion recognition. A rise of 2-9% is seen in specificity after 3D face restoration.

Table 5.9. Performance on Occlusion Recognition

Metrics	**Accuracy**	**Sensitivity**	**Specificity**
Dataset	Before 3D Face Restoration		
Bosphorus	90.5	94.7	78.2
UMBDB	77.8	85.2	65.9
KinectFaceDB	82.7	89.3	75.4
	After 3D Face Restoration		
Bosphorus	94.2	97.3	80.6
UMBDB	85.8	93.9	74.6
KinectFaceDB	90.6	95.1	79.3

5.5 Summary

In this chapter, 3D landmark-based face restoration is proposed for face recognition. The proposed technique utilizes the concept of Variational Autoencoders (VAE), triplet loss, and support vector machine (SVM). Support vector machine achieves the final classification of faces. Medium gaussian SVM is used for prediction model after the deep learning phase. VAE produces optimised deep features from the restored landmarks. Triplet loss is used for enhanced embedding. The proposed approach has been compared with three recently developed approaches over three well-known datasets. The experimental results reveal that the proposed approach attained an 8-10% improvement over the existing techniques. The computation time obtained from the proposed approach is approximately 50% faster than the other techniques. Further, in ablation studies, the performance of the proposed model is tested for occluded versus non-occluded faces, gender, emotion, and occlusion recognition. The experimental results reveal that the proposed technique is able to recognize the occluded face with gender classification.

6. Conclusions and Future Scope

In this chapter, the outcomes of each chapter have been highlighted. The future scope of the presented work has also been discussed.

6.1 Conclusions

With the advent of deep learning and the current COVID-19 pandemic times, there is a need of contact-less high-quality biometrics system that can handle the requirements. Face recognition is one of the no-physical-contact biometric system. 3D face recognition system has been worked up on since more than a decade now. A lot of research using deep learning has been done for 3D face reconstruction, designing facial caricatures, facial puppetry, face re-enactment, the virtual 3D makeup invariant of facial expressions, and face aging. The contribution of this thesis are follows:

- 3D face recognition is one of an effective contactless method of biometric recognition. Face recognition using mesh images, RGB-D, voxels, or the point cloud images comes under the category of 3D face recognition. There were multiple challenges in 2D face recognition viz. illumination, occlusion, make-up, etc. The face reconstruction problem solves the problem of incomplete information about face and enriches the model with more information.
- From the existing literature, it has been found that the 3D face restoration techniques are still a challenging research area due to non-availability of large 3D public face datasets and practical constraints of high power GPU requirements for processing millions of 3D faces in an economical way. Most techniques are using 3D images as 2D image with time or annotated 2D face. Some of the techniques are using depth images which are 2.5D images and not purely 3D. These techniques suffer from low quality face generation when restoration is applied. The facial details are missing such as texture and shape. The major drawbacks of the existing 3D face reconstruction techniques are that very few approaches have been built using 3D images. Another drawback is the lack of publicly available 3D face dataset with millions of 3D faces using voxel, mesh, or RGB-D image gives the benchmark of 3D face recognition and reconstruction a scope of improvement over the

coming years.

- To overcome this problem, a novel technique for voxel-based 3D occlusion-invariant face recognition using game theory and simulated annealing has been proposed. The proposed technique is capable of achieving better results in minimum time. The adversarial triplet loss was used to train the model based on generator and discriminator. This technique is one of the first to use generative model on voxels. The 3D face recognition system has the capabilities to handle occlusion-invariance. Tthe overall classification accuracy obtained from voxel-based facial recognition is 86.1%. For occlusion invariant facial recognition, the average accuracy generated by the proposed technique is 75.5%. In the case of facial recognition using 3D landmarks, the average accuracy achieved from the given methodology is 81.3%. In the case of 3D mesh face recognition, the average precision achieved from the methodology is 83.9%. Adversarial generation technique for triplet generation promotes minimal bias. The technique coupled with simulated annealing allows the proposed system to be resilient in a variety of areas using voxels. The drawback of the technique is that it is not an end-to-end deep learning technique. The process involves a pipeline of multiple techniques which increases the overall complexity of the model at training time.
- A novel technique for voxel-based 3D face reconstruction using sequential deep learning has been developed. The conversion of the mesh based 3D image to voxelized form is done. The voxels were further used to recover the missing voxel information using the mirroring technique. The pipeline for 3D face recognition is the recovered 3D face voxels were converted to deep feature embeddings by passing them through variational autoencoders, bidirectional long short term memory, and finally the triplet loss training for getting the desired classification. In the ablation studies, various experimentations have been performed for gender recognition, emotion recognition, occlusion recognition, and finally the face recognition. In terms of gender identification, the accuracy is greater than 90% following restoration for all three datasets due to binary classification. In emotional identification, the accuracy for Bosphorus dataset is 94.57% after restoration as compared to 83.95% before restoration, i.e., 10.62% improvement in accuracy. In the case of University of Milano Bicocca 3D face database and Kinect Face database, there is a rise of 14.9% and 12.49%. In occlusion classification, there is a rise of 4.07%, 4.82%, and 7.05%

for Bosphorus, University of Milano Bicocca 3D face database, and Kinect Face databases, respectively. The precision of the proposed architecture (3DVFR) is state-of-the-art with 90.01% in 3D face recognition using Bosphorus dataset voxels. In the case of University of Milano Bicocca 3D face database and Kinect Face database, the performance of the facial recognition is 78.21% and 85.68%, respectively. The limitation of the approach is the scalability problem. This technique has been tested on datasets having less than 5K images in each. If the same technique is tested on 1.8M image 4DFAB dataset, there will be a need of hyper-parameter in case of deep learning based models. Ensembling technique has been applied in the model, which increases the time complexity of the model resulting in need of more resources.

- A novel technique based on the 3D landmark-based face restoration technique using the variational autoencoders and triplet loss training has been proposed. The 3D facial landmarks were used for building the reconstruction technique based on mirroring technique and mid-face plane. The experimental findings indicate that the proposed method obtained an 8-10 % increase over the current techniques. The computation period derived from the proposed method is roughly 50% quicker than the other approaches. Furthermore, in ablation experiments, the efficiency of the proposed model is checked for occluded versus non-occluded faces, gender, emotion, and occlusion type recognition. The limitation of this technique is that it will not be able to generated a high-quality textured 3D face image. This technique depends on the accurate labelling of the 3D facial landmarks manually or automatically. This technique is a pipeline based technique and not an end-to-end deep learning technique which results in higher time complexity of the model while training.

6.2 Comparison of Proposed Techniques

In this section, the comparison of the proposed techniques [87], [146], [147] of Chapter 3, 4, and 5 have been done in Table 6.1.

Table 6.1. Detailed comparison of proposed techniques

Year	Author	Approaches/Models	Pros	Cons
2020	Sharma and Kumar [146]	Deep learning process, game-theory based generator and discriminator	- End to end deep learning technique. - Occlusion invariant reconstruction is done.	- Not tested for being scalable.
2020	Sharma and Kumar [87]	Variational autoencoder, bi-LSTM, and triplet loss train-ing	- Mirroring technique is faster for reconstruction as compared to deep learning-based methods. - The preprocessing, reconstruction, and verification time are faster in computation as compared to state-of-the-art.	- Preprocessing does not include facial alignment. - Technique has not been tested on a big dataset of 3D faces.
2021	Sharma and Kumar [147]	Variational autoencoders and triplet loss training	- Landmarks based reconstruction is faster as compared to mesh-based reconstruction using the mirroring technique.	- Face alignment is missing. - Deep learning model can be trained better if the dataset is huge.

6.3 Future Scope

The present work is the use of 3D face reconstruction for face recognition, emotion, gender, and occlusion recognition. Some directions with regard to the future work are documented below:

- This work can be extended for 4D face recognition, by adding the time factor into the datasets. A dataset namely, 4DFAB has already been generated by Imperial College London. This dataset was launched in CVPR 2018, and it has still not been made public.
- 3D face reconstruction can also be helpful in 3D animation movies, especially in the augmented as well as virtual reality based environments.
- The lip reconstruction as well as face swapping for 3D images can be utilized for 3D face recognition and reconstruction techniques.
- The generative adversarial networks can be used for reconstruction of 3D faces based on the available facial data in mesh, voxel, point cloud, or RGB-D format.
- The proposed techniques in the thesis namely, V3DOFR, 3DVFR, and 3DLFR can be extended for 4D face recognition and reconstruction.

List of Publications

Papers Published

1. Sharma, S. and Kumar, V., 2021. 3D Landmark-based Face Restoration for Recognition using Variational Autoencoder and Triplet Loss. *IET Biometrics*, 10(1), pp. 87-98. [**I.F. 2.588**]. SCI.
2. Sharma, S. and Kumar, V., 2021. Performance evaluation of machine learning based face recognition techniques. Wireless Personal Communications, 118, pp. 3403-3433. [**I.F.: 1.671**]. SCI.
3. Sharma, S. and Kumar, V., 2020. Voxel-based 3D face reconstruction and its application to face recognition using sequential deep learning. *Multimedia Tools and Applications*, 79, pp.17303-17330. [**I.F. 2.757**]. SCI.
4. Sharma, S. and Kumar, V., 2020. Voxel-based 3D occlusion-invariant face recognition using game theory and simulated annealing. *Multimedia Tools and Applications*, 79(35), pp. 26517-26547 [**I.F. 2.757**]. SCI
5. Sharma, S. and Kumar, V., 2018. Performance evaluation of 2D face recognition techniques under image processing attacks. *Modern Physics Letters B*, 32(19), pp.1850212 (1-22). [**I.F. 1.668**]. SCI.
6. Sharma, S. and Kumar, V., 2020. Low-Level Features based 2D Face Recognition using Machine Learning. International Journal of Intelligent Engineering Informatics. Inderscience Publishers, 8(4). E-SCI.

Papers under Review

1. Sharma, S. and Kumar, V., 2021. 3D Face Reconstruction in Deep Learning Era: A Survey.

Books Chapters Published

1. Sharma, S. and Kumar, V., 2019. Transfer Learning in 2.5 D Face Image for Occlusion Presence and Gender Classification. In Handbook of Research on Deep Learning Innovations and Trends (pp. 97-113). IGI Global.

Bibliography

[1] X. Zhang, L. Yin, J.F. Cohn, S. Canavan, M. Reale, A. Horowitz, P. Liu, and J.M. Girard, "BP4D-Spontaneous: A high-resolution spontaneous 3D dynamic facial expression database," Image Vis. Comput., vol. 32, no. 10, pp. 692–706, 2014, doi: 10.1016/j.imavis.2014.06.002.

[2] X. Zhang, , L. Yin, J.F. Cohn, S. Canavan, M. Reale, A. Horowitz, and P. Liu, "A High-Resolution Spontaneous 3D Dynamic Facial Expression Database." IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), pp. 1-6, 2013.

[3] T. C. Faltemier, K. W. Bowyer, and P. J. Flynn, "Using multi-instance enrollment to improve performance of 3D face recognition," Comput. Vis. Image Underst., vol. 112, no. 2, pp. 114–125, Nov. 2008, doi: 10.1016/j.cviu.2008.01.004.

[4] S. Sharma and V. Kumar, "Performance evaluation of 2D face recognition techniques under image processing attacks," Mod. Phys. Lett. B, vol. 32, no. 19, p. 1850212, Jul. 2018, doi: 10.1142/S0217984918502123.

[5] S. Sharma and V. Kumar, "Transfer Learning in 2.5D Face Image for Occlusion Presence and Gender Classification," pp. 97–113, 2019, doi: 10.4018/978-1-5225-7862-8.ch006.

[6] Y. Xu, M. Gong, H. Fu, D. Tao, K. Zhang, and K. Batmanghelich, "Multi-scale masked 3-D U-Net for brain tumor segmentation," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Sep. 2019, vol. 11384 LNCS, pp. 222–233, doi: 10.1007/978-3-030-11726-9_20.

[7] E. Stoykova, A. Ayd, P. Benzie, N. Grammalidis, S. Malassiotis, J. Ostermann, S. Piekh, V. Sainov, C. Theobalt, T. Thevar, and X. Zabulis, "3-D time-varying scene capture technologies - A survey," IEEE Transactions on Circuits and Systems for Video Technology, vol. 17, no. 11. pp. 1568–1585, Nov. 2007, doi: 10.1109/TCSVT.2007.909975.

[8] H. Patil, A. Kothari, and K. Bhurchandi, "3-D face recognition: features, databases, algorithms and challenges," Artif. Intell. Rev., vol. 44, no. 3, pp. 393–441, Oct. 2015, doi: 10.1007/s10462-015-9431-0.

[9] D. Kim, M. Hernandez, J. Choi, and G. Medioni, "Deep 3D face identification," IEEE Int.

Jt. Conf. Biometrics, IJCB 2017, vol. 2018-Janua, pp. 133–142, 2018, doi: 10.1109/BTAS.2017.8272691.

[10] A. K. Maurya and A. K. Tripathi, "ECP: a novel clustering-based technique to schedule precedence constrained tasks on multiprocessor computing systems," Computing, vol. 101, no. 8, pp. 1015–1039, Aug. 2019, doi: 10.1007/s00607-018-0636-3.

[11] J. Geng, T. Shao, Y. Zheng, Y. Weng, and K. Zhou, "Warp-guided GANs for single-photo facial animation," Dec. 2018, doi: 10.1145/3272127.3275043.

[12] M. Sankaranarayanan, C. Mala, and S. Mathew, "Pre-processing framework with virtual mono-layer sequence of boxes for video based vehicle detection applications," Multimed. Tools Appl., pp. 1–28, Sep. 2020, doi: 10.1007/s11042-020-09587-x.

[13] Y. Liu, G. Zhao, B. Gong, Y. Li, R. Raj, N. Goel, S. Kesav, S. Gottimukkala, Z. Wang, W. Ren, and D. Tao, "Image Dehazing: Improved Techniques," in Deep Learning Through Sparse and Low-Rank Modeling, Elsevier, 2019, pp. 251–262.

[14] A. Pumarola, A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer, "GANimation: Anatomically-aware Facial Animation from a Single Image." In Proceedings of the European conference on computer vision (ECCV), pp. 818-833, 2018.

[15] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2Face: Real-time Face Capture and Reenactment of RGB Videos." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2387-2395, 2016.

[16] S. K. Pandey, R. B. Mishra, and A. K. Tripathi, "BPDET: An effective software bug prediction model using deep representation and ensemble learning techniques," Expert Syst. Appl., vol. 144, p. 113085, Apr. 2020, doi: 10.1016/j.eswa.2019.113085.

[17] A. S. Rajput, B. Raman, and J. Imran, "Privacy-preserving human action recognition as a remote cloud service using RGB-D sensors and deep CNN," Expert Syst. Appl., vol. 152, p. 113349, Aug. 2020, doi: 10.1016/j.eswa.2020.113349.

[18] V. Chouhan, S.K. Singh, A. Khamparia, D. Gupta, P. Tiwari, C. Moreira, R. Damaševičius, and V.H.C De Albuquerque, "A novel transfer learning based approach for pneumonia detection in chest X-ray images," Appl. Sci., vol. 10, no. 2, Jan. 2020, doi: 10.3390/app10020559.

[19] L. Jiang, J. Zhang, B. Deng, H. Li, and L. Liu, "3D Face Reconstruction with Geometry Details from a Single Image," IEEE Trans. Image Process., vol. 27, no. 10, pp. 4756–4770,

Oct. 2018, doi: 10.1109/TIP.2018.2845697.

[20] M. Feng, S. Z. Gilani, Y. Wang, and A. Mian, “3D face reconstruction from light field images: A model-free approach,” Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 11214 LNCS, pp. 508–526, 2018, doi: 10.1007/978-3-030-01249-6_31.

[21] G. Bhatt, P. Jha, and B. Raman, “Representation learning using step-based deep multi-modal autoencoders,” Pattern Recognit., vol. 95, pp. 12–23, Nov. 2019, doi: 10.1016/j.patcog.2019.05.032.

[22] E. Richardson, M. Sela, and R. Kimmel, “3D face reconstruction by learning from synthetic data,” in Proceedings - 2016 4th International Conference on 3D Vision, 3DV 2016, Dec. 2016, pp. 460–467, doi: 10.1109/3DV.2016.56.

[23] P. Dou, S. K. Shah, and I. A. Kakadiaris, “End-to-end 3D face reconstruction with deep neural networks,” Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017, vol. 2017-Janua, pp. 1503–1512, 2017, doi: 10.1109/CVPR.2017.164.

[24] S. Z. Gilani, A. Mian, and P. Eastwood, “Deep, dense and accurate 3D face correspondence for generating population specific deformable models,” Pattern Recognit., vol. 69, pp. 238–250, Sep. 2017, doi: 10.1016/j.patcog.2017.04.013.

[25] X. H. Xiao and G. Leedham, “Signature verification by neural networks with selective attention,” Appl. Intell., vol. 11, no. 2, pp. 213–223, Sep. 1999, doi: 10.1023/A:1008380515294.

[26] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, B. C. Van Esesn, A. A. S. Awwal, and V. K. Asari, “The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches,” Mar. 2018, Accessed: Oct. 13, 2020. [Online]. Available: http://arxiv.org/abs/1803.01164.

[27] M. Z. Alom, T.M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M.S. Nasrin, B.C. Van Esesn, A.A.S Awwal, and V.K. Asari, “A state-of-the-art survey on deep learning theory and architectures,” Electronics (Switzerland), vol. 8, no. 3. MDPI AG, Mar. 01, 2019, doi: 10.3390/electronics8030292.

[28] B. Devkota, A. Alsadoon, P. W. C. Prasad, A. K. Singh, and A. Elchouemi, “Image Segmentation for Early Stage Brain Tumor Detection using Mathematical Morphological Reconstruction,” in Procedia Computer Science, Jan. 2018, vol. 125, pp. 115–123, doi:

10.1016/j.procs.2017.12.017.

[29] D. Yang, A. Alsadoon, P. W. C. Prasad, A. K. Singh, and A. Elchouemi, "An Emotion Recognition Model Based on Facial Recognition in Virtual Learning Environment," in Procedia Computer Science, Jan. 2018, vol. 125, pp. 2–10, doi: 10.1016/j.procs.2017.12.003.

[30] V. Nguyen, V. Nguyen, M. Blumenstein, V. Muthukkumarasamy, and G. Leedham, "Off-line Signature Verification Using Enhanced Modified Direction Features in Conjunction with Neural Classifiers and Support Vector Machines," Accessed: Oct. 19, 2020. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.323.6490.

[31] J. Pantaleoni, "VoxelPipe: A Programmable Pipeline for 3D Voxelization," In Proceedings of the ACM SIGGRAPH Symposium on High Performance Graphics, pp. 99-106, 2011.

[32] C. J. Ogáyar, A. J. Rueda, R. J. Segura, and F. R. Feito, "Fast and simple hardware accelerated voxelizations using simplicial coverings," Vis. Comput., vol. 23, no. 8, pp. 535–543, Aug. 2007, doi: 10.1007/s00371-007-0097-8.

[33] L. Olanrewaju, O. Oyebiyi, S. Misra, R. Maskeliunas, and R. Damasevicius, "Secure ear biometrics using circular kernel principal component analysis, Chebyshev transform hashing and Bose–Chaudhuri–Hocquenghem error-correcting codes," Signal, Image Video Process., vol. 14, no. 5, pp. 847–855, Jul. 2020, doi: 10.1007/s11760-019-01609-y.

[34] N. Sapkota, A. Alsadoon, P. W. C. Prasad, A. Elchouemi, and A. K. Singh, "Data Summarization Using Clustering and Classification: Spectral Clustering Combined with k-Means Using NFPH," in Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Prespectives and Prospects, COMITCon 2019, Feb. 2019, pp. 146–151, doi: 10.1109/COMITCon.2019.8862218.

[35] X. He and P. Niyogi, "Locality Preserving Projections." In Advances in neural information processing systems, (pp. 153-160), 2004.

[36] M. Belkin and P. Niyogi, "Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering.", In Advances in neural information processing systems, pp. 585-591, 2002.

[37] F. Schroff and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 815-823, 2015.

[38] I. Goodfellow, "NIPS 2016 Tutorial: Generative Adversarial Networks," pp. 1-57, Dec.

2016, [Online]. Available: http://arxiv.org/abs/1701.00160.

[39] B. Gecer, S. Ploumpis, I. Kotsia, and S. Zafeiriou, “Ganfit: Generative adversarial network fitting for high fidelity 3D face reconstruction,” Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2019-June, pp. 1155–1164, 2019, doi: 10.1109/CVPR.2019.00125.

[40] D. Yu, G. Hinton, N. Morgan, J. T. Chien, and S. Sagayama, “Introduction to the special section on deep learning for speech and language processing,” IEEE Transactions on Audio, Speech and Language Processing, vol. 20, no. 1. pp. 4–6, 2012, doi: 10.1109/TASL.2011.2173371.

[41] Y. Konig and N. Morgan, “GDNN: a gender-dependent neural network for continuous speech recognition,” Jan. 2003, pp. 332–337, doi: 10.1109/ijcnn.1992.226966.

[42] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative Adversarial Nets.”, In Advances in neural information processing systems (pp. 2672-2680), 2014. [Online]. Available: http://www.github.com/goodfeli/adversarial.

[43] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved Techniques for Training GANs.”, In Advances in neural information processing systems, pp. 2234-2242, 2016. [Online]. Available: https://github.com/openai/improved-gan.

[44] Z. He, W. Zuo, S. Member, M. Kan, S. Shan, and X. Chen, “AttGAN: Facial Attribute Editing by Only Changing What You Want.”, IEEE Transactions on Image Processing, 28(11), pp.5464-5478, 2019. [Online]. Available: http://arxiv.org/abs/1711.10678.

[45] N. Rajagopalan and C. Mala, “Optimization of QoS parameters for channel allocation in cellular networks using soft computing techniques”, in Advances in Intelligent and Soft Computing, 2012, vol. 130 AISC, no. VOL. 1, pp. 621–631, doi: 10.1007/978-81-322-0487-9_60.

[46] G. Antipov, M. Baccouche, and J.-L. Dugelay, “FACE AGING WITH CONDITIONAL GENERATIVE ADVERSARIAL NETWORKS”, IEEE international conference on image processing (ICIP), pp. 2089-2093, 2017.

[47] A. Boesen, L. Larsen, S. K. Sønderby, H. Larochelle, O. Winther, and O. Dk, “Autoencoding beyond pixels using a learned similarity metric”, In International

conference on machine learning, pp. 1558-1566. PMLR. 2016.

[48] G. Perarnau, J. van de Weijer, B. Raducanu, and J. M. Álvarez, “Invertible Conditional GANs for image editing,” Nov. 2016, [Online]. Available: http://arxiv.org/abs/1611.06355.

[49] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, “Optimization by Simulated Annealing,” Science (80-.)., vol. 220, no. 4598, pp. 671 LP – 680, May 1983, doi: 10.1126/science.220.4598.671.

[50] S. Bandyopadhyay, U. Maulik, and M. K. Pakhira, “11:23 WSPC/115-IJPRAI,” 2001. [Online]. Available: www.worldscientific.com.

[51] R. Caves, S. Quegan, and R. White, “Quantitative Comparison of the Performance of SAR Segmentation Algorithms”, IEEE Transactions on Image Processing, 7(11), pp.1534-1546, 1998.

[52] U. Maulik, S. Bandyopadhyay, and J. C. Trinder, “SAFE: An Efficient Feature Extraction Technique”, Knowledge and information systems, 3(3), pp.374-387, 2001

[53] S. Bandyopadhyay, S. Saha, U. Maulik, and K. Deb, “A simulated annealing-based multiobjective optimization algorithm: AMOSA,” IEEE Trans. Evol. Comput., vol. 12, no. 3, pp. 269–283, 2008, doi: 10.1109/TEVC.2007.900837.

[54] D. P. Kingma and M. Welling, “Auto-Encoding Variational Bayes,” Dec. 2013, [Online]. Available: http://arxiv.org/abs/1312.6114.

[55] D. J. C. MacKay, Information theory, inference, and learning algorithms. Cambridge University Press, 2003.

[56] A. Graves and J. Schmidhuber, “Framewise phoneme classification with bidirectional LSTM and other neural network architectures,” in Neural Networks, Jul. 2005, vol. 18, no. 5–6, pp. 602–610, doi: 10.1016/j.neunet.2005.06.042.

[57] B. Heiselet, P. Hoj, and T. Poggio, “Face Recognition with Support Vector Machines: Global versus Component-based Approach” In Proceedings Eighth IEEE International Conference on Computer Vision. ICCV, Vol. 2, pp. 688-694, 2001.

[58] E. U. Haq, X. Huarong, and M. I. Khattak, “Face recognition by SVM using local binary patterns,” in Proceedings - 2017 14th Web Information Systems and Applications Conference, WISA 2017, Apr. 2018, vol. 2018-January, pp. 172–175, doi: 10.1109/WISA.2017.68.

[59] B. Schölkopf, K.K. Sung, C.J. Burges, F. Girosi, P. Niyogi, T. Poggio and V. Vapnik,

"Comparing Support Vector Machines with Gaussian Kernels to Radial Basis Function Classifiers", IEEE transactions on Signal Processing, 45(11), pp.2758-2765, 1997.

[60] B. Egger, W.A. Smith, A. Tewari, S. Wuhrer, M. Zollhoefer, T. Beeler, F. Bernard, T. Bolkart, A. Kortylewski, S. Romdhani, and C. Theobalt, "3D Morphable Face Models—Past, Present, and Future," ACM Trans. Graph., vol. 39, no. 5, pp. 1–38, 2020, doi: 10.1145/3395208.

[61] V. Blanz and T. Vetter, "Face Recognition Based on Fitting a 3D Morphable Model", IEEE Transactions on pattern analysis and machine intelligence, 25(9), pp.1063-1074, 2003.

[62] J. Booth, A. Roussos, A. Ponniah, D. Dunaway, and S. Zafeiriou, "Large Scale 3D Morphable Models," Int. J. Comput. Vis., vol. 126, no. 2–4, pp. 233–254, 2018, doi: 10.1007/s11263-017-1009-7.

[63] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou, "FaceWarehouse: A 3D facial expression database for visual computing," IEEE Trans. Vis. Comput. Graph., vol. 20, no. 3, pp. 413–425, Mar. 2014, doi: 10.1109/TVCG.2013.249.

[64] T. Gerig, A. Morel-Forster, C. Blumer, B. Egger, M. Luthi, S. Schonborn, and T. Vetter, "Morphable face models - An open framework," Proc. - 13th IEEE Int. Conf. Autom. Face Gesture Recognition, FG 2018, pp. 75–82, 2018, doi: 10.1109/FG.2018.00021.

[65] P. Huber, G. Hu, R. Tena, P. Mortazavian, P. Koppen, W.J. Christmas, M. Ratsch, and J. Kittler, "A Multiresolution 3D Morphable Face Model and Fitting Framework", In Proceedings of the 11th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, 2016. doi: 10.5220/0005669500790086.

[66] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero, "Learning a model of facial shape and expression from 4D scans," in ACM Transactions on Graphics, Nov. 2017, vol. 36, no. 6, pp. 1–17, doi: 10.1145/3130800.3130813.

[67] J. Lin, Y. Yuan, T. Shao, and K. Zhou, "Towards High-Fidelity 3D Face Reconstruction From In-the-Wild Images Using Graph Convolutional Networks," pp. 5890–5899, 2020, doi: 10.1109/cvpr42600.2020.00593.

[68] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3D face model for pose and illumination invariant face recognition," in 6th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2009, 2009, pp. 296–301, doi: 10.1109/AVSS.2009.58.

[69] H. Kim, M. Zollhöfer, A. Tewari, J. Thies, C. Richardt, and C. Theobalt, “InverseFaceNet: Deep Monocular Inverse Face Rendering”, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4625-4634, 2018.

[70] A. S. Jackson, A. Bulat, V. Argyriou, and G. Tzimiropoulos, “Large Pose 3D Face Reconstruction from a Single Image via Direct Volumetric CNN Regression,” Proc. IEEE Int. Conf. Comput. Vis., vol. 2017-Octob, pp. 1031–1039, 2017, doi: 10.1109/ICCV.2017.117.

[71] D. Eigen, C. Puhrsch, and R. Fergus, “Depth Map Prediction from a Single Image using a Multi-Scale Deep Network.”, In Advances in neural information processing systems, pp. 2366-2374, 2014.

[72] A. Saxena, S. H. Chung, and A. Y. Ng, “3-D depth reconstruction from a single still image,” Int. J. Comput. Vis., vol. 76, no. 1, pp. 53–69, Jan. 2008, doi: 10.1007/s11263-007-0071-y.

[73] S. Tulsiani, T. Zhou, A. A. Efros, and J. Malik, “Multi-view Supervision for Single-view Reconstruction via Differentiable Ray Consistency”, in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2626-2634, 2017. [Online]. Available: https://shubhtuls.github.io/.

[74] M. Tatarchenko, A. Dosovitskiy, and T. Brox, “Octree Generating Networks: Efficient Convolutional Architectures for High-resolution 3D Outputs.” In Proceedings of the IEEE International Conference on Computer Vision, pp. 2088-2096, 2017. [Online]. Available: https://github.com/lmb-freiburg/ogn.

[75] E. Richardson, M. Sela, R. Or-El, and R. Kimmel, “Learning detailed face reconstruction from a single image,” Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017, vol. 2017-Janua, pp. 5553–5562, 2017, doi: 10.1109/CVPR.2017.589.

[76] J. Roth, Y. Tong, and X. Liu, “Adaptive 3D Face Reconstruction from Unconstrained Photo Collections”, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4197-4206, 2016.

[77] I. Kemelmacher-Shlizerman and S. M. Seitz, “Face reconstruction in the wild,” in Proceedings of the IEEE International Conference on Computer Vision, 2011, pp. 1746–1753, doi: 10.1109/ICCV.2011.6126439.

[78] I. Kemelmacher-Shlizerman and R. Basri, “3D face reconstruction from a single image using a single reference face shape,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no.

2, pp. 394–405, 2011, doi: 10.1109/TPAMI.2010.63.

[79] Z. Zhu, P. Luo, X. Wang, and X. Tang, “Deep learning identity-preserving face space,” in Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 113–120, doi: 10.1109/ICCV.2013.21.

[80] Y. Tang, R. Salakhutdinov, and G. Hinton, “Deep Lambertian Networks,” 2012.

[81] J. Lehtinen, “Facial Performance Capture with Deep Neural Networks Samuli Laine NVIDIA Tero Karras NVIDIA Timo Aila NVIDIA Antti Herva Remedy Entertainment.”

[82] Z. Liu, P. Luo, X. Wang, and X. Tang, “Deep Learning Face Attributes in the Wild” In Proceedings of the IEEE international conference on computer vision, pp. 3730-3738, 2015.

[83] V. Nair, J. Susskind, and G. E. Hinton, “Analysis-by-Synthesis by Learning to Invert Generative Black Boxes”, In International Conference on Artificial Neural Networks, pp. 971-981, Springer, Berlin, Heidelberg, 2008.

[84] X. Peng, R. S. Feris, X. Wang, and D. N. Metaxas, “A Recurrent Encoder-Decoder Network for Sequential Face Alignment,” Aug. 2016, Accessed: Oct. 13, 2020. [Online]. Available: http://arxiv.org/abs/1608.05477.

[85] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks.” Accessed: Oct. 13, 2020. [Online]. Available: http://code.google.com/p/cuda-convnet/.

[86] W. Zhu, N. Zeng, and N. Wang, “Sensitivity, Specificity, Accuracy, Associated Confidence Interval and ROC Analysis with Practical SAS ® Implementations”, NESUG proceedings: health care and life sciences, Baltimore, Maryland, 19, pp.67, 2010.

[87] S. Sharma and V. Kumar, “Voxel-based 3D face reconstruction and its application to face recognition using sequential deep learning,” Multimed. Tools Appl., vol. 79, no. 25–26, pp. 17303–17330, Jul. 2020, doi: 10.1007/s11042-020-08688-x.

[88] “Understanding Categorical Cross-Entropy Loss, Binary Cross-Entropy Loss, Softmax Loss, Logistic Loss, Focal Loss and all those confusing names.” https://gombru.github.io/2018/05/23/cross_entropy_loss/ (accessed Oct. 15, 2020).

[89] “Difference Between Categorical and Sparse Categorical Cross Entropy Loss Function – LeakyReLU.” https://leakyrelu.com/2020/01/01/difference-between-categorical-and-sparse-categorical-cross-entropy-loss-function/ (accessed Oct. 15, 2020).

[90] B. L. Kalman and S. C. Kwasny, “Why tanh: choosing a sigmoidal function,” Jan. 2003, pp. 578–581, doi: 10.1109/ijcnn.1992.227257.

[91] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” Dec. 2014, [Online]. Available: http://arxiv.org/abs/1412.6980.

[92] S. Bouaziz, Y. Wang, and M. Pauly, “Online modeling for realtime facial animation,” ACM Trans. Graph., vol. 32, no. 4, pp. 1–10, Jul. 2013, doi: 10.1145/2461912.2461976.

[93] C. Cao, Q. Hou, and K. Zhou, “Displaced dynamic expression regression for real-time facial tracking and animation,” in ACM Transactions on Graphics, Jul. 2014, vol. 33, no. 4, pp. 1–10, doi: 10.1145/2601097.2601204.

[94] M. Zollhöfer, J. Thies, P. Garrido, D. Bradley, T. Beeler, P. Pérez, M. Stamminger, M. Nießner, and C. Theobalt, “State of the art on monocular 3D face reconstruction, tracking, and applications,” Comput. Graph. Forum, vol. 37, no. 2, pp. 523–550, 2018, doi: 10.1111/cgf.13382.

[95] J. Thies, M. Elgharib, A. Tewari, C. Theobalt, and M. Nießner, “Neural Voice Puppetry: Audio-driven Facial Reenactment,” pp. 1–23, 2019, [Online]. Available: http://arxiv.org/abs/1912.05566.

[96] P. Garrido, Valgaerts, L., Sarmadi, H., Steiner, I., Varanasi, K., Perez, P. and Theobalt, C., “VDub: Modifying Face Video of Actors for Plausible Visual Alignment to a Dubbed Audio Track,” Comput. Graph. Forum, vol. 34, no. 2, pp. 193–204, May 2015, doi: 10.1111/cgf.12552.

[97] P. Garrido, L. Valgaerts, C. Wu, and C. Theobalt, “Reconstructing detailed dynamic face geometry from monocular video,” ACM Trans. Graph., vol. 32, no. 6, pp. 1–10, Nov. 2013, doi: 10.1145/2508363.2508380.

[98] C. Siegl, V. Lange, M. Stamminger, F. Bauer, and J. Thies, “FaceForge: Markerless Non-Rigid Face Multi-Projection Mapping”, IEEE transactions on visualization and computer graphics, 23(11), pp.2440-2446.

[99] F. Maninchedda, M. R. Oswald, and M. Pollefeys, “Fast 3D reconstruction of faces with glasses,” Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017, vol. 2017-Janua, no. 2, pp. 4608–4617, 2017, doi: 10.1109/CVPR.2017.490.

[100] S. Zhang, H. Yu, T. Wang, L. Qi, J. Dong, and H. Liu, “Dense 3D facial reconstruction from a single depth image in unconstrained environment,” Virtual Real., vol. 22, no. 1, pp.

37–46, 2018, doi: 10.1007/s10055-017-0311-6.

[101] L. Jiang, X. Wu, and J. Kittler, “Pose Invariant 3D Face Reconstruction,” pp. 1–8, 2018, [Online]. Available: http://arxiv.org/abs/1811.05295.

[102] F. Wu, S. Li, T. Zhao, K. N. Ngan, and L. Sheng, “Cascaded regression using landmark displacement for 3D face reconstruction,” Pattern Recognit. Lett., vol. 125, pp. 766–772, Jul. 2019, doi: 10.1016/j.patrec.2019.07.017.

[103] D. Kollias, S. Cheng, E. Ververas, I. Kotsia, and S. Zafeiriou, “Deep Neural Network Augmentation: Generating Faces for Affect Analysis,” Int. J. Comput. Vis., vol. 128, no. 5, pp. 1455–1484, 2020, doi: 10.1007/s11263-020-01304-3.

[104] Cheng, S., Kotsia, I., Pantic, M. and Zafeiriou, S., “4DFAB: A Large Scale 4D Facial Expression Database for Biometric Applications | DeepAI.” In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5117-5126, 2018.

[105] J. Lyu, X. Li, X. Zhu, and C. Cheng, “Pixel-Face: A Large-Scale, High-Resolution Benchmark for 3D Face Reconstruction,” 2020, [Online]. Available: http://arxiv.org/abs/2008.12444.

[106] G. Anbarjafari, R. E. Haamer, I. LÜSi, T. Tikk, and L. Valgma, “3D face reconstruction with region based best fit blending using mobile phone for virtual reality based social media,” Bull. Polish Acad. Sci. Tech. Sci., vol. 67, no. 1, pp. 125–132, 2019, doi: 10.24425/bpas.2019.127341.

[107] Y. Xing, R. Tewari, and P. R. S. Mendonça, “A self-supervised bootstrap method for single-image 3D face reconstruction,” Proc. - 2019 IEEE Winter Conf. Appl. Comput. Vision, WACV 2019, pp. 1014–1023, 2019, doi: 10.1109/WACV.2019.00113.

[108] S. Zulqarnain Gilani and A. Mian, “Learning from Millions of 3D Scans for Large-Scale 3D Face Recognition,” Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., pp. 1896–1905, 2018, doi: 10.1109/CVPR.2018.00203.

[109] X. Li, G. Hu, J. Zhu, W. Zuo, M. Wang, and L. Zhang, “Learning Symmetry Consistent Deep CNNs for Face Completion,” IEEE Trans. Image Process., vol. 29, pp. 7641–7655, 2020, doi: 10.1109/TIP.2020.3005241.

[110] X. Han, K. Hou, D. Du, Y. Qiu, S. Cui, K. Zhou, and Y. Yu, “CaricatureShop: Personalized and Photorealistic Caricature Sketching,” IEEE Trans. Vis. Comput. Graph., vol. 26, no. 7, pp. 2349–2361, 2020, doi: 10.1109/TVCG.2018.2886007.

[111] S. Moschoglou, S. Ploumpis, M. A. Nicolaou, A. Papaioannou, and S. Zafeiriou, "3DFaceGAN: Adversarial Nets for 3D Face Representation, Generation, and Translation," Int. J. Comput. Vis., 2020, doi: 10.1007/s11263-020-01329-8.

[112] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition", In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778, 2016. [Online]. Available: http://image-net.org/challenges/LSVRC/2015/.

[113] F. Liu, D. Zeng, J. Li, and Q. jun Zhao, "On 3D face reconstruction via cascaded regression in shape space," Front. Inf. Technol. Electron. Eng., vol. 18, no. 12, pp. 1978–1990, Dec. 2017, doi: 10.1631/FITEE.1700253.

[114] A. Tewari, M. Zollhofer, H. Kim, P. Garrido, F. Bernard, P. Perez, and C. Theobalt, "MoFA: Model-Based Deep Convolutional Face Autoencoder for Unsupervised Monocular Reconstruction," Proc. - 2017 IEEE Int. Conf. Comput. Vis. Work. ICCVW 2017, vol. 2018-Janua, pp. 1274–1283, 2017, doi: 10.1109/ICCVW.2017.153.

[115] "Visual Geometry Group - University of Oxford." http://www.robots.ox.ac.uk/~vgg/data/vgg_face/ (accessed Oct. 13, 2020).

[116] X. Han, C. Gao, and Y. Yu, "DeepSketch2Face: A deep learning based sketching system for 3D face and caricature modeling," ACM Trans. Graph., vol. 36, no. 4, pp. 1–12, 2017, doi: 10.1145/3072959.3073629.

[117] G. S. Hsu, H. C. Shie, C. H. Hsieh, and J. S. Chan, "Fast Landmark Localization with 3D Component Reconstruction and CNN for Cross-Pose Recognition," IEEE Trans. Circuits Syst. Video Technol., vol. 28, no. 11, pp. 3194–3207, Nov. 2018, doi: 10.1109/TCSVT.2017.2748379.

[118] X. Cao, Z. Chen, A. Chen, X. Chen, S. Li, and J. Yu, "Sparse Photometric 3D Face Reconstruction Guided by Morphable Models," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., pp. 4635–4644, 2018, doi: 10.1109/CVPR.2018.00487.

[119] A. T. Tran, T. Hassner, I. Masi, E. Paz, Y. Nirkin, and G. Medioni, "Extreme 3D Face Reconstruction: Seeing Through Occlusions," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., pp. 3935–3944, 2018, doi: 10.1109/CVPR.2018.00414.

[120] Z. H. Feng, P. Huber, J. Kittler, P. Hancock, X.J. Wu, Q. Zhao, P. Koppen, and M. Rätsch, "Evaluation of dense 3D reconstruction from 2D face images in the wild," Proc. - 13th IEEE

Int. Conf. Autom. Face Gesture Recognition, FG 2018, pp. 780–786, Jun. 2018, doi: 10.1109/FG.2018.00123.

[121] Y. Feng, F. Wu, X. Shao, Y. Wang, and X. Zhou, “Joint 3d face reconstruction and dense alignment with position map regression network,” Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 11218 LNCS, pp. 557–574, 2018, doi: 10.1007/978-3-030-01264-9_33.

[122] F. Liu, R. Zhu, D. Zeng, Q. Zhao, and X. Liu, “Disentangling Features in 3D Face Shapes for Joint Face Reconstruction and Recognition,” Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., pp. 5216–5225, 2018, doi: 10.1109/CVPR.2018.00547.

[123] N. Chinaev, A. Chigorin, and I. Laptev, “MobileFace: 3D face reconstruction with efficient CNN regression,” Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 11132 LNCS, pp. 15–30, 2019, doi: 10.1007/978-3-030-11018-5_3.

[124] Y. Deng, J. Yang, S. Xu, D. Chen, Y. Jia, and X. Tong, “Accurate 3D face reconstruction with weakly-supervised learning: From single image to image set,” IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work., vol. 2019-June, pp. 285–295, 2019, doi: 10.1109/CVPRW.2019.00038.

[125] X. Yuan and I. K. Park, “Face de-occlusion using 3D morphable model and generative adversarial network,” Proc. IEEE Int. Conf. Comput. Vis., vol. 2019-Octob, pp. 10061–10070, 2019, doi: 10.1109/ICCV.2019.01016.

[126] Y. Luo, X. Tu, and M. Xie, “Learning Robust 3D Face Reconstruction and Discriminative Identity Representation,” 2019 2nd IEEE Int. Conf. Inf. Commun. Signal Process. ICICSP 2019, pp. 317–321, 2019, doi: 10.1109/ICICSP48821.2019.8958506.

[127] B. Gecer, A. Lattas, S. Ploumpis, J. Deng, A. Papaioannou, S. Moschoglou, and S. Zafeiriou, “Synthesizing Coupled 3D Face Modalities by Trunk-Branch Generative Adversarial Networks,” pp. 1–19, 2019, [Online]. Available: http://arxiv.org/abs/1909.02215.

[128] Y. Chen, F. Wu, Z. Wang, Y. Song, Y. Ling, and L. Bao, “Self-supervised Learning of Detailed 3D Face Reconstruction,” pp. 1–11, 2019, [Online]. Available: http://arxiv.org/abs/1910.11791.

[129] “Large-scale CelebFaces Attributes (CelebA) Dataset.”

http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html (accessed Oct. 13, 2020).

[130] “Labelled Faces in the Wild (LFW) Dataset | Kaggle.” https://www.kaggle.com/jessicali9530/lfw-dataset (accessed Oct. 13, 2020).

[131] W. Ren, J. Yang, S. Deng, D. Wipf, X. Cao, and X. Tong, “Face video deblurring using 3D facial priors,” Proc. IEEE Int. Conf. Comput. Vis., vol. 2019-Octob, pp. 9387–9396, 2019, doi: 10.1109/ICCV.2019.00948.

[132] X. Tu, J. Zhao, M. Xie, Z. Jiang, A. Balamurugan, Y. Luo, Y. Zhao, L. He, Z. Ma, and J. Feng, “3D Face Reconstruction from A Single Image Assisted by 2D Face Images in the Wild,” IEEE Trans. Multimed., 2020, doi: 10.1109/TMM.2020.2993962.

[133] A. Jourabloo and X. Liu, “Pose-Invariant 3D Face Alignment.” pp. 3694–3702, 2015.

[134] “(No Title).” https://arxiv.org/pdf/1712.01443.pdf (accessed Oct. 14, 2020).

[135] F. Liu, Q. Zhao, X. Liu, and D. Zeng, “Joint Face Alignment and 3D Face Reconstruction with Application to Face Recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, no. 3, pp. 664–678, Mar. 2020, doi: 10.1109/TPAMI.2018.2885995.

[136] Z. Ye, R. Yi, M. Yu, J. Zhang, Y.-K. Lai, and Y. Liu, “3D-CariGAN: An End-to-End Solution to 3D Caricature Generation from Face Photos,” no. Figure 1, pp. 1–17, 2020, [Online]. Available: http://arxiv.org/abs/2003.06841.

[137] J. Huo, W. Lia, Y. Shia, Y. Gaoa, and H. Yinb, “WebCaricature: A Benchmark for Caricature Face Recognition.” https://cs.nju.edu.cn/rl/WebCaricature.htm (accessed Oct. 14, 2020).

[138] A. Lattas, S. Moschoglou, B. Gecer, S. Ploumpis, V. Triantafyllou, A. Ghosh and S. Zafeiriou, “AvatarMe: Realistically Renderable 3D Facial Reconstruction ‘In-the-Wild,’” pp. 757–766, 2020, doi: 10.1109/cvpr42600.2020.00084.

[139] J. Zhang, H. Cai, Y. Guo, and Z. Peng, “Landmark Detection and 3D Face Reconstruction for Caricature using a Nonlinear Parametric Model,” pp. 1–12, 2020, [Online]. Available: http://arxiv.org/abs/2004.09190.

[140] Y. Deng, J. Yang, D. Chen, F. Wen, and X. Tong, “Disentangled and Controllable Face Image Generation via 3D Imitative-Contrastive Learning,” pp. 5153–5162, 2020, doi: 10.1109/cvpr42600.2020.00520.

[141] K. Li, J. Yang, N. Jiao, J. Zhang, and Y.-K. Lai, “Adaptive 3D Face Reconstruction from a Single Image,” pp. 1–11, 2020, [Online]. Available: http://arxiv.org/abs/2007.03979.

[142] B. Chaudhuri, N. Vesdapunt, L. Shapiro, and B. Wang, “Personalized Face Modeling for Improved Face Reconstruction and Motion Retargeting,” pp. 1–20, 2020, [Online]. Available: http://arxiv.org/abs/2007.06759.

[143] J. Shang, T. Shen, S. Li, L. Zhou, M. Zhen, T. Fang, and L. Quan, “Self-Supervised Monocular 3D Face Reconstruction by Occlusion-Aware Multi-view Geometry Consistency,” 2020, [Online]. Available: http://arxiv.org/abs/2007.12494.

[144] X. Cai, H. Yu, J. Lou, X. Zhang, G. Li, and J. Dong, “3D Facial Geometry Recovery from a Depth View with Attention Guided Generative Adversarial Network,” Sep. 2020.

[145] S. Xu, J. Yang, D. Chen, F. Wen, Y. Deng, Y. Jia, and X. Tong, “Deep 3D Portrait From a Single Image,” pp. 7707–7717, 2020, doi: 10.1109/cvpr42600.2020.00773.

[146] S. Sharma and V. Kumar, Voxel-based 3D occlusion-invariant face recognition using game theory and simulated annealing, vol. 79, no. 35–36. Multimedia Tools and Applications, 2020.

[147] S. Sharma and V. Kumar, “3D landmark - based face restoration for recognition using variational autoencoder and triplet loss,” vol. 10(1), no. August 2020, pp. 87–98, 2021, doi: 10.1049/bme2.12005.

[148] J. Zhang, L. Lin, J. Zhu, and S. C. H. Hoi, “Weakly-Supervised Multi-Face 3D Reconstruction,” pp. 1–9, 2021, [Online]. Available: http://arxiv.org/abs/2101.02000.

[149] M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof, “Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization,” Proc. IEEE Int. Conf. Comput. Vis., pp. 2144–2151, 2011, doi: 10.1109/ICCVW.2011.6130513.

[150] “ICG - AFLW.” https://www.tugraz.at/institute/icg/research/team-bischof/lrs/downloads/aflw/ (accessed Oct. 14, 2020).

[151] S. Moschoglou, A. Papaioannou, C. Sagonas, J. Deng, I. Kotsia, and S. Zafeiriou, “AgeDB: the first manually collected, in-the-wild age database.” Accessed: Oct. 14, 2020. [Online]. Available: https://ibug.doc.ic.ac.uk/resources/agedb/.

[152] “Morphace.” https://faces.dmi.unibas.ch/bfm/main.php?nav=1-1-0&id=details (accessed Oct. 14, 2020).

[153] A. Savran, N. Alyüz, H. Dibeklioğlu, O. Çeliktutan, B. Gökberk, B. Sankur, and L. Akarun, " Bosphorus database for 3D face analysis.", In European workshop on biometrics and identity management, pp. 47-56, 2008.

[154] “3D Facial Expression Database - Binghamton University.” http://www.cs.binghamton.edu/~lijun/Research/3DFE/3DFE_Analysis.html (accessed Oct. 13, 2020).

[155] “Center for Biometrics and Security Research.” http://www.cbsr.ia.ac.cn/english/3DFace Databases.asp (accessed Oct. 14, 2020).

[156] D. Yi, Z. Lei, S. Liao, and S. Z. Li, “Learning Face Representation from Scratch,” Nov. 2014, [Online]. Available: http://arxiv.org/abs/1411.7923.

[157] “Celebrities in Frontal-Profile in the Wild.” http://www.cfpw.io/ (accessed Oct. 14, 2020).

[158] H. Yang, H. Zhu, Y. Wang, M. Huang, Q. Shen, R. Yang, and X. Cao, “FaceScape: a Large-scale High Quality 3D Face Dataset and Detailed Riggable 3D Face Prediction,” pp. 598–607, Mar. 2020, Accessed: Oct. 16, 2020. [Online]. Available: http://arxiv.org/abs/2003.13989.

[159] “FaceWarehouse.” http://kunzhou.net/zjugaps/facewarehouse/ (accessed Oct. 13, 2020).

[160] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, "Overview of the face recognition grand challenge" In 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05) (Vol. 1, pp. 947-954). IEEE..

[161] MORENO and A., “GavabDB : A 3d face database,” Proc. 2nd COST275 Work. Biometrics Internet, 2004, pp. 75–80, 2004, Accessed: Oct. 13, 2020. [Online]. Available: https://ci.nii.ac.jp/naid/10028232899.

[162] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang, “Interactive facial feature localization,” Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 7574 LNCS, no. PART 3, pp. 679–692, 2012, doi: 10.1007/978-3-642-33712-3_49.

[163] “IJB-A Dataset Request Form | NIST.” https://www.nist.gov/itl/iad/image-group/ijb-dataset-request-form (accessed Oct. 14, 2020).

[164] R. Min, N. Kose, and J. L. Dugelay, “KinectfaceDB: A kinect database for face recognition,” IEEE Trans. Syst. Man, Cybern. Syst., vol. 44, no. 11, pp. 1534–1548, Nov. 2014, doi: 10.1109/TSMC.2014.2331215.

[165] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar, “Localizing parts of faces using a consensus of exemplars,” Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern

Recognit., pp. 545–552, 2011, doi: 10.1109/CVPR.2011.5995602.

[166] A. Bulat and G. Tzimiropoulos, “How far are we from solving the 2D & 3D Face Alignment problem? (and a dataset of 230,000 3D facial landmarks).” [Online]. Available: www.adrianbulat.com/face-alignment/.

[167] A. D. Bagdanov, A. Del Bimbo, and I. Masi, “The florence 2D/3D hybrid face dataset,” in Proceedings of the 2011 joint ACM workshop on Human gesture and behavior understanding - J-HGBU ’11, 2011, p. 79, doi: 10.1145/2072572.2072597.

[168] “Notre Dame CVRL.” https://cvrl.nd.edu/projects/data/#nd-2006-data-set (accessed Oct. 13, 2020).

[169] “Laboratory for Image and Video Engineering - The University of Texas at Austin.” http://live.ece.utexas.edu/research/texas3dfr/ (accessed Oct. 14, 2020).

[170] H. A. Le and I. A. Kakadiaris, “UHDB31: A Dataset for Better Understanding Face Recognition Across Pose and Illumination Variation,” in 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), Oct. 2017, vol. 2018-Janua, pp. 2555–2563, doi: 10.1109/ICCVW.2017.300.

[171] A. Colombo, C. Cusano, and R. Schettini, “UMB-DB: A database of partially occluded 3D faces,” in Proceedings of the IEEE International Conference on Computer Vision, 2011, pp. 2113–2119, doi: 10.1109/ICCVW.2011.6130509.

[172] O. M. Parkhi, A. Vedaldi, and A. Zisserman, “Deep Face Recognition.”

[173] C. Sanderson, “The VidTIMIT Database,” 2002. [Online]. Available: http://www.idiap.ch.

[174] J. Son Chung, A. Nagrani, and A. Zisserman, “VoxCeleb2: Deep Speaker Recognition.” Accessed: Oct. 14, 2020. [Online]. Available: www.robots.ox.ac.uk/.

[175] “YouTube Faces Database : Main.” https://www.cs.tau.ac.il/~wolf/ytfaces/ (accessed Oct. 14, 2020).

[176] “300-VW | Computer Vision Online.” https://computervisiononline.com/dataset/1105138793 (accessed Oct. 13, 2020).

[177] “i·bug - resources - 300 Faces In-the-Wild Challenge (300-W), ICCV 2013.” https://ibug.doc.ic.ac.uk/resources/300-W/ (accessed Oct. 14, 2020).

[178] V. Vijayan, K. Bowyer, and P. Flynn, “3D twins and expression challenge,” in Proceedings of the IEEE International Conference on Computer Vision, 2011, pp. 2100–2105, doi: 10.1109/ICCVW.2011.6130507.

[179] A. Ranjan, T. Bolkart, S. Sanyal, and M. J. Black, “Generating 3D faces using Convolutional Mesh Autoencoders.” [Online]. Available: http://coma.is.tue.mpg.de/.

[180] Y. Zhao, Z. Jin, G. Qi, H. Lu, and X. Hua, “An Adversarial Approach to Hard Triplet Generation.”

[181] T.-T. Do, T. Tran, I. Reid, V. Kumar, T. Hoang, and G. Carneiro, “A Theoretically Sound Upper Bound on the Triplet Loss for Improving the Efficiency of Deep Distance Metric Learning.”

[182] A. ElSayed, E. Kongar, A. Mahmood, T. Sobh, and T. Boult, “Neural Generative Models for 3D Faces with Application in 3D Texture Free Face Recognition,” Nov. 2018, [Online]. Available: http://arxiv.org/abs/1811.04358.

[183] Y. Tan, H. Lin, Z. Xiao, S. Ding, and H. Chao, “Face Recognition from Sequential Sparse 3D Data via Deep Registration,” in 2019 International Conference on Biometrics (ICB), Jun. 2019, pp. 1–8, doi: 10.1109/ICB45273.2019.8987284.

[184] Y. Xu, X. Zhu, Z. Li, G. Liu, Y. Lu, and H. Liu, “Using the original and ‘symmetrical face’ training samples to perform representation based two-step face recognition,” Pattern Recognit., vol. 46, no. 4, pp. 1151–1158, Apr. 2013, doi: 10.1016/j.patcog.2012.11.003.

[185] Y. Xu, X. Li, J. Yang, and D. Zhang, “Integrate the original face image and its mirror image for face recognition,” Neurocomputing, vol. 131, pp. 191–199, May 2014, doi: 10.1016/j.neucom.2013.10.025.

[186] Y. Xu, X. Fang, X. Li, J. Yang, J. You, H. Liu, and S. Teng, “Data uncertainty in face recognition,” IEEE Trans. Cybern., vol. 44, no. 10, pp. 1950–1961, Oct. 2014, doi: 10.1109/TCYB.2014.2300175.

[187] J. C. Bezdek, “FCM: THE FUZZY c-MEANS CLUSTERING ALGORITHM,” 1984.

[188] S. Singh and S. S. Kasana, “Efficient classification of the hyperspectral images using deep learning,” Multimed. Tools Appl., vol. 77, no. 20, pp. 27061–27074, Oct. 2018, doi: 10.1007/s11042-018-5904-x.

[189] D. Celis and M. Rao, “Learning facial recognition biases through VAE latent representations,” in FAT/MM 2019 - Proceedings of the 1st International Workshop on Fairness, Accountability, and Transparency in MultiMedia, co-located with MM 2019, Oct. 2019, pp. 26–32, doi: 10.1145/3347447.3356752.

[190] X. Zhou, J. Lin, J. Jiang, and S. Chen, “Learning a 3D gaze estimator with improved itracker

combined with bidirectional LSTM," in Proceedings - IEEE International Conference on Multimedia and Expo, Jul. 2019, vol. 2019-July, pp. 850–855, doi: 10.1109/ICME.2019.00151.

[191] G. Tian, Y. Yuan, and Y. Liu, "Audio2Face: Generating speech/face animation from single audio with attention-based bidirectional LSTM networks," in Proceedings - 2019 IEEE International Conference on Multimedia and Expo Workshops, ICMEW 2019, Jul. 2019, pp. 366–371, doi: 10.1109/ICMEW.2019.00069.

[192] H. Li and H. Xu, "Video-Based Sentiment Analysis with hvnLBP-TOP Feature and bi-LSTM," Proc. AAAI Conf. Artif. Intell., vol. 33, no. 01, pp. 9963–9964, Jul. 2019, doi: 10.1609/aaai.v33i01.33019963.

[193] C. Huang, Y. Li, C. L. Chen, and X. Tang, "Deep Imbalanced Learning for Face Recognition and Attribute Prediction," IEEE Trans. Pattern Anal. Mach. Intell., pp. 1–1, May 2019, doi: 10.1109/tpami.2019.2914680.

[194] H. H. Tsai and Y. C. Chang, "Facial expression recognition using a combination of multiple facial features and support vector machine," Soft Comput., vol. 22, no. 13, pp. 4389–4405, Jul. 2018, doi: 10.1007/s00500-017-2634-3.

[195] B. Richhariya and D. Gupta, "Facial expression recognition using iterative universum twin support vector machine," Appl. Soft Comput. J., vol. 76, pp. 53–67, Mar. 2019, doi: 10.1016/j.asoc.2018.11.046.

[196] V. K. Verma, S. Srivastava, T. Jain, and A. Jain, "Local invariant feature-based gender recognition from facial images," in Advances in Intelligent Systems and Computing, vol. 817, Springer Verlag, 2019, pp. 869–878.

[197] N. B. Kar, K. S. Babu, A. K. Sangaiah, and S. Bakshi, "Face expression recognition system based on ripplet transform type II and least square SVM," Multimed. Tools Appl., vol. 78, no. 4, pp. 4789–4812, Feb. 2019, doi: 10.1007/s11042-017-5485-0.

[198] Y. D. Y. Zhang, Y. D. Y. Zhang, X. X. Hou, H. Chen, and S. H. Wang, "Seven-layer deep neural network based on sparse autoencoder for voxelwise detection of cerebral microbleed," Multimed. Tools Appl., pp. 1–18, Mar. 2017, doi: 10.1007/s11042-017-4554-8.

[199] M. Sultan Zia, M. Hussain, and M. Arfan Jaffar, "A novel spontaneous facial expression recognition using dynamically weighted majority voting based ensemble classifier,"

Multimed. Tools Appl., vol. 77, no. 19, pp. 25537–25567, Oct. 2018, doi: 10.1007/s11042-018-5806-y.

[200] Y. Xiao, J. Wu, Z. Lin, and X. Zhao, "A deep learning-based multi-model ensemble method for cancer prediction," Comput. Methods Programs Biomed., vol. 153, pp. 1–9, Jan. 2018, doi: 10.1016/j.cmpb.2017.09.005.

[201] L. Yu, R. Zhou, L. Tang, and R. Chen, "A DBN-based resampling SVM ensemble learning paradigm for credit classification with imbalanced data," Appl. Soft Comput. J., vol. 69, pp. 192–202, Aug. 2018, doi: 10.1016/j.asoc.2018.04.049.

[202] G. Thiesen, B. F. Gribel, and M. P. M. Freitas, "Facial asymmetry: A current review," Dental Press J. Orthod., vol. 20, no. 6, pp. 110–125, Nov. 2015, doi: 10.1590/2177-6709.20.6.110-125.sar.

[203] M. Alqattan, J. Djordjevic, A. I. Zhurov, and S. Richmond, "Comparison between landmark and surface-based three-dimensional analyses of facial asymmetry in adults," Eur. J. Orthod., vol. 37, no. 1, pp. 1–12, Feb. 2015, doi: 10.1093/ejo/cjt075.

[204] M. T. Liu, R.A. Iglesias, S.S. Sekhon, Y. Li, K. Larson, A. Totonchi, and B. Guyuron, "Factors contributing to facial asymmetry in identical twins," Plast. Reconstr. Surg., vol. 134, no. 4, pp. 638–646, 2014, doi: 10.1097/PRS.0000000000000554.

[205] I. Ercan, S.T. Ozdemir, A. Etoz, D. Sigirli, R.S. Tubbs, M. Loukas, and I. Guney, "Facial asymmetry in young healthy subjects evaluated by statistical shape analysis," J. Anat., vol. 213, no. 6, pp. 663–669, Dec. 2008, doi: 10.1111/j.1469-7580.2008.01002.x.

[206] O. E. Linden, J. Kit He, C. S. Morrison, S. R. Sullivan, and H. O. B. Taylor, "The relationship between age and facial asymmetry," in Plastic and Reconstructive Surgery, 2018, vol. 142, no. 5, pp. 1145–1152, doi: 10.1097/PRS.0000000000004831.

[207] A. Rikhtegar, M. Pooyan, and M. Taghi Manzuri-Shalmani, "GA-OPTIMIZED STRUCTURE OF CNN FOR FACE RECOGNITION APPLICATIONS IET Review Copy Only IET Computer Vision."

[208] K. Sundararajan and D. L. Woodard, "Deep learning for biometrics: A survey," ACM Computing Surveys, vol. 51, no. 3. Association for Computing Machinery, Apr. 01, 2018, doi: 10.1145/3190618.

[209] P. Korshunov and S. Marcel, "DeepFakes: a New Threat to Face Recognition? Assessment and Detection," Dec. 2018, [Online]. Available: http://arxiv.org/abs/1812.08685.

[210] S. Han, H. Mao, and W. J. Dally, “Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding,” Oct. 2015, [Online]. Available: http://arxiv.org/abs/1510.00149.

[211] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive Growing of GANs for Improved Quality, Stability, and Variation,” Oct. 2017, [Online]. Available: http://arxiv.org/abs/1710.10196.