



68 landmarks are efficient for 3D face alignment: what about more?

3D face alignment method applied to face recognition

Marwa Jabberi^{1,2} · Ali Wali² · Bidyut Baran Chaudhuri³ · Adel M. Alimi^{2,4}

Received: 2 September 2021 / Revised: 29 April 2022 / Accepted: 4 February 2023 /

Published online: 1 April 2023

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract

This paper proposes a 3D face alignment of 2D face images in the wild with noisy landmarks. The objective is to recognize individuals from their single profile image. We first proceed by extracting more than 68 landmarks using a bag of features. This allows us to obtain a bag of visible and invisible facial keypoints. Then, we reconstruct a 3D face model and get a triangular mesh by meshing the obtained keypoints. For each face, the number of keypoints is not the same, which makes this step very challenging. Later, we process the 3D face using butterfly and BPA algorithms to make correlation and regularity between 3D face regions. Indeed, 2D-to-3D annotations give much higher quality to the 3D reconstructed face model without the need for any additional 3D Morphable models. Finally, we carry out alignment and pose correction steps to get frontal pose by fitting the rendered 3D reconstructed face to 2D face and performing pose normalization to achieve good rates in face recognition. The recognition step is based on deep learning and it is performed using DCNNs, which are very powerful and modern, for feature learning and face identification. To verify the proposed method, three popular benchmarks, YTF, LFW, and BIWI databases, are tested. Compared to the best recognition results reported on these benchmarks, our proposed method achieves comparable or even better recognition performances.

✉ Marwa Jabberi
marwa.jabberi@regim.usf.tn

Ali Wali
ali.wali@isims.usf.tn

Bidyut Baran Chaudhuri
bbc@isical.ac.in

Adel M. Alimi
adel.alimi@regim.usf.tn

¹ University of Sousse, ISITCom, 4011 Sousse, Tunisia

² REsearch Groups in Intelligent Machines (REGIM Lab), University of Sfax, National Engineering School of Sfax (ENIS), BP 1173, Sfax 3038, Tunisia

³ Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata 700108, India

⁴ Department of Electrical and Electronic Engineering Science, Faculty of Engineering and the Built Environment, University of Johannesburg, Johannesburg, South Africa

Keywords 3D face recognition · 3D face alignment · Deep learning · DCNNs · Feature extraction · 3D mesh reconstruction · 3D mesh preprocessing

1 Introduction

Face Recognition (FR) is a common authentication tool in many applications. The face has a physical biometric characteristic that is non-invasive and is accepted by users. There is no direct contact with the acquisition device as required when using the iris and the finger print. Nowadays, FR is the most widespread technique used in authentication [1]. The face does not characterize only humans, also animals have facial features [38, 40]. The main architecture of FR is shown in the figure below (Fig. 1).

The classical FR pipeline consists of two phases: Online and Offline phases. In the Offline phase, the user is not logged in. Part of the dataset follows the preprocessing steps, such as face cropping, denoising, smoothing, and alignment. The feature extraction step is carried out to compute the biometric signature of each face, then to classify these signatures to categorize the features. However, the Online phase is carried out at each interrogation of the dataset by the user, where a query face goes through the same steps; i.e., preprocessing and feature extraction. Then, a check is established to know the belonging rate for each class and to establish a related decision. The decision step is a 1:N problem that compares a query face image using its biometric computed signature against all the stored signatures to determine the identity of the query face. The run time of this phase must be reduced.

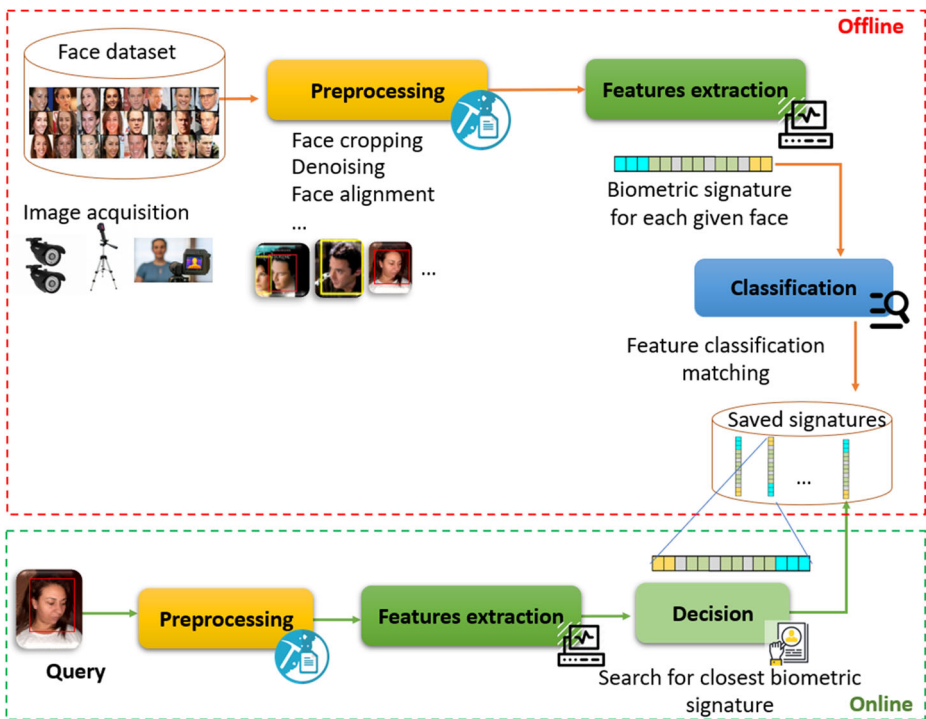


Fig. 1 Classical face recognition pipeline

The challenges facing FR are the lighting conditions, pose variation, occlusions, facial expressions, and low resolution [66]. All these problems decrease the recognition rate. To solve them, the preprocessing step must be well-developed to strengthen the recognition system and to achieve good results in recognizing the face as taken in the wild. Numerous approaches related to this field have been advanced; however, several challenges still persist [7, 23].

In this paper, we present a method in the context of FR. The aim is to tackle the pose variation problem because recognizing and identifying a person from a single 2D image under pose variation remains a great challenge. To get the highest recognition rate, an alignment step has to be well-developed. It is an essential preprocessing step in face recognition. Thus, our work includes:

1. Feature extraction to add keypoints to the 68 traditional fiducial landmarks since these keypoints provide rich information about facial geometry.
2. 3D face reconstruction from 2D obtained keypoints of a single image under an arbitrary view to localize the self-occluded face parts in the case of large poses.
3. 3D face alignment by fitting 3D reconstructed face to 2D face image using keypoints marching to render the frontal view by pose normalization and correction.
4. Application of face recognition using Deep Convolutional Neural Networks (DCNNs) on the aligned faces.

Indeed, facial alignment and 3D reconstruction are two different tasks. Currently, the relationship between these two tasks has become known. Indeed, 2D face alignment has shown weakness consisting in its inability to address large poses. The relationship between 3D face reconstruction and face alignment consists essentially in mapping and estimating the 3D face geometry from a single 2D image. The main objective is to compute the visibility and position of 2D landmarks.

Recent methods have used hand-crafted features to improve performance, especially for the earliest contributions. In this paper, our approach is applied directly to RGB face images using compact features with engineered descriptors to achieve good performance. The power of CNNs, which are used to learn the features on large multi-identities datasets for 3D face alignment with application to face recognition, is therefore exploited.

2 Related works

2.1 Face recognition

Currently, FR is a widely used biometric technique since the face has become the most attractive biometric. Also, the COVID-19 pandemic has changed several statistics worldwide, including the biometric modalities. In the earliest results of FindBiometrics (Fig. 2) reported in a review survey [30], FR has retained the top spot as the year's most used and exciting modality.

2.1.1 Face Recognition studies

FR methods can be classified into three categories: global also known as holistic methods using the entire facial surface [3], local methods based on local regions or patches and not considering the whole face [51], and hybrid methods [74] consisting in combining global and local feature descriptors.

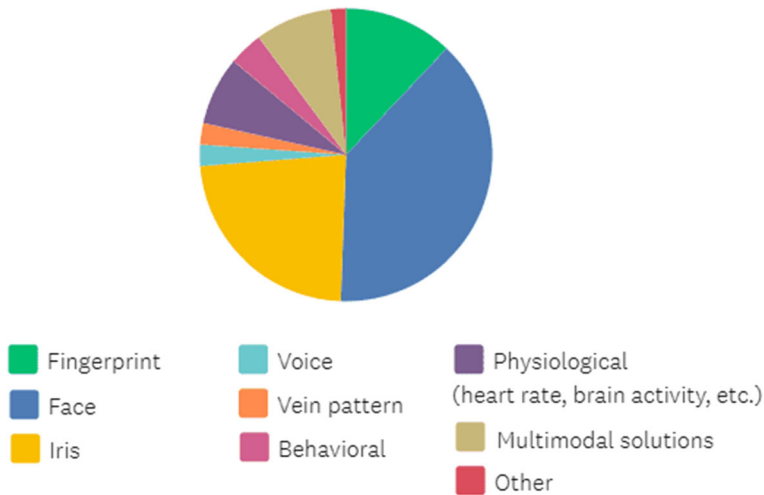


Fig. 2 Results of the latest FindBiometrics until year 2020

Global Face Recognition methods The global or holistic methods for 2D FR extract features from the entire facial surface. The used descriptors are not dedicated to give information on a specific part or region of the face. Indeed, extracted features give information on the entire facial surface. It is time consuming but very efficient to synthesize the complete face.

EigenFace [88] is a global FR method which uses Principal Component Analysis (PCA) [72]. Eigen Vectors are measured to describe faces, which are computed by measuring the features of the nose tip, mouth, eye corners, and chin edges. Since global methods project face representation into a small subspace or a correlation plane, EigenFaces are projected onto a reduced face space by PCA. Eigenface has been used in several other works accompanied by modifications and improvements as presented in [67].

Fisherface [43] is part of the known holistic method in FR whose principle is based on maximizing the separation between classes during training. Fisherface reduces face space dimension using PCA. Also, Fisher's Linear Discriminant (FDL) [95] method is used to generate face features as a linear combination able to separate two or more classes. This famous algorithm has undergone several modifications in several criteria as presented in [34, 73].

In [75], a comparative study between Fisherface and EigenFace is presented. Other methods focusing on fusion based on PCA and FDL are also presented in [69].

Joshua et al. [86] presented a non linear holistic approach capable of extracting complex natural observations and ensuring a global optimal solution of the true structure convergence of face images under low dimensional input-space.

LaplacianFaces [33] consists in mapping the face into a face subspace based on Locality Preserving Projections (LPP) [32] to get the best global face description.

Local Face Recognition methods FR methods based on local features focus on fiducial points and parts of the face to generate features. These techniques compute the local features through pixel parameters, face histograms, geometric shapes, and correlation planes between different regions. Local feature-based methods require no face representation reduction since it is a work on local features of the face.

The most popular used techniques are based on different descriptors, such as Local Binary Pattern (LBP) and its derivatives [47], Histogram of Oriented Gradients (HOG) [55], Vander Lugt Correlator (VLC), Scale Invariant Feature Transform (SIFT). All these descriptors are presented in [78].

In [16], Chandrakala et al. dealt with pose variations, scale, facial expression, and illumination challenges using a cascading of LBP and HOG.

A recent work presented in [46] is based on a variation of the Local Radius of Gyration Face (LRGF), invariant to lighting conditions variation, pose change, and noise.

Hybrid Face Recognition methods Hybrid FR methods consist of a fusion of global and local methods. In fact, the global characteristics are combined with the local ones, making this FR category the most efficient and robust [27, 65].

In a recent work [24], the authors focused on features optimization by the selection of the optimal characteristics of the face with Particle Swarm Optimization (PSO) algorithm based on the face active region of interest.

A FR system using the LBP Histogram (LBPH) descriptor for local and global special features of the face is presented in [21].

Table 1 presents a brief study of some FR approaches.

2.1.2 Deep face recognition studies

With the advent of BigData and Data Mining, methods and approaches for FR have become numerous. In this work, our goal is to recognize individuals from their faces under pose variations using CNNs. This method proved to have impressive results. With the advent of CPU and GPU cores [54], CNNs and Deep CNNs have been used in a huge number of training data.

CNNs can be classified among the category of hybrid FR methods. They are adapted to feature learning and label prediction, as well as to mapping the input data to deep features, which are the output of the last hidden layer. They are later to the predicted labels. Feature learning is carried out automatically and it is shared as weights between different layers. However, DCNNs achieve superior performance since they are able to extract high level features ensured by the classification architecture [93]. Once deep features are extracted, most of the methods directly calculate the similarity between the two features using cosine, L2, or the nearest neighbor (NN) metrics, and therefore establish comparison for identification. Yet, deep networks which perform perfectly on benchmark datasets may fail in real world applications.

Most of the recent methods perform face image representation using hand-crafted local image descriptors, such as SIFT, LBP, and HOG [9, 48, 61].

Contrary to the aforementioned methods, our method is applied to RGB pixels without combining other descriptors to improve performance.

Researchers have used CNNs and DCNNs in FR application, either for features learning, features extraction, or features classification.

In CosFace [94], Large Margin Cosine Loss (LMCL), as a novel loss function, is performed to remove radial variations and to maximize the decision margin in the angular space. LMCL guides DCNNs to learn the highly discriminative face features. So, intra-class variance is minimized and inter-class variance is maximized.

SphereFace [58] represents class centers in the angular space and penalizes the angles between deep features and their corresponding weights in a multiplicative way, since authors

Table 1 Summary of some FR methods in the literature in a chronological order

Author/Year	Facial features	Treated challenge	Observations and main results
Liu et al. [57]	Eigenface Fisherface	Expression	Asymmetry measures are invariant to facial expressions.
Napoléon et al. [64]	LBP VLC DOG	Scale Illumination	Correlation optimization against illumination changes by descriptors adaptation.
Huang et al. [35]	Local Vector Pattern (LVP) feature-point Bilateral Recognition (BR)	—	Geometrical Model generated from feature points can provide important information for FR.
Vinary et al. [90]	SIFT SURF	Scale	Pertinent mathematical arguments based on SIFT and SURF features are robust in unconstrained scenarios of FR
Arigbabu et al. [4]	Laplacian filter and Pyramid HOG	Pose	Local feature descriptors are robust to photometric variations.
Zhang et al. [98]	Chromaticity Intrinsic Image (CII)	Illumination	Building a reflectance model for skin surface, lighting source, and camera sensor interactions Face color appearance elaboration.
Maliha et al. [62]	Haar cascade EigenFace FisherFace LBP	—	Facial features space reduction with local features combination.
Muqheet et al. [50]	Interpolation-based directional wavelet transform (DIWT) LBP	Pose Expression	Implementation of quadtree partitioning for local regions extraction from face images to obtain DIWT combined with LBP features.
Gupta et al. [31]	Various combinations of SIFT, SURF and LPP features	Illumination Scale	Low false-positive rate serves for high accurate face detection.

Table 2 Summary of some Deep-based FR methods in the literature in a chronological order

Author/Year	Used Technique	Model	Treated Challenge	Observations and main results
Sun et al. [80]	HOG features based on arc-cosine kernel similarity between images measurement	Deep match kernels (DMK)	–	High inter-class ambiguity and similarity measurement and huge intra-class variability.
Rao et al. [68]	Metric learning and adversarial learning features combination	MTCNN Discriminative aggregation network DAN	Pose	Network aggregation produces more discriminative synthesized images.
Cui et al. [19]	Depth estimation using fully convolutional network (FCN)	Inception V2	Pose Expression Occlusion	Face depth estimation improves 2D FR.
Cao et al. [15]	Texture warping controlled by Adversarial Residual Dictionary Learning (ARDL) and dense correspondence to bind face surfaces	Multilayer CNN	Pose Resolution	Identity information and texture details preservation lead to realistic high-resolution results.
Shi et al. [76]	Pose correction based on 3DMM fitting Histogram-based features learning	VGG-Face	Pose	Pose-invariant FR.
Zou et al. [102]	Multi-feature fusion framework Gabor transformation	Small-scale adaptive deep CNN model	Pose	Identification system based on multi-level feature fusion solves pose variation problem.

found out that linear transformation matrix in the last fully connected layer of the CNN is useful for this issue. Thus, an Additive Angular Margin Loss helps to obtain the highly discriminative features learned via DCNNs for FR.

In the same context, RegularFace [99] uses intuitive geometric interpretation by penalizing the angle between an identity and its nearest neighbor by focusing on intra-class compactness.

In [56], the authors focused on decreasing information redundancy in features learning and on maintaining the most informative components of spatial feature maps. This module, called attention, is added to the convolutional layer of a standard CNN.

FR methods based on deep CNNs are in full development. Indeed, to have a high recognition rate, it is absolutely necessary to focus on features since CNNs perform feature learning in an automatic way. So, most methods add a module or an additional function to CNNs layers or focus on the preprocessing steps to keep only the salient features of the face (Table 2).

2.2 Face alignment

As mentioned in the previous subsection, the recognition rate is relative to the extracted and the learned face features. For this reason, the face must be well preprocessed before performing the recognition test.

The alignment process forms a part of the preprocessing steps and involves the placement of the face in a frontal position (pitch (ϕ) = 0° , yaw (γ) = 0° and roll (θ) = 0°). More precisely, it is pose normalization since the frontal pose covers the canonical view of the face taken arbitrarily in the wild. Aligning poses make FR easier.

In the majority of papers, authors refer to face alignment as face detection while aligning faces consists in establishing a rotation in the plane and making the face in a frontal view.

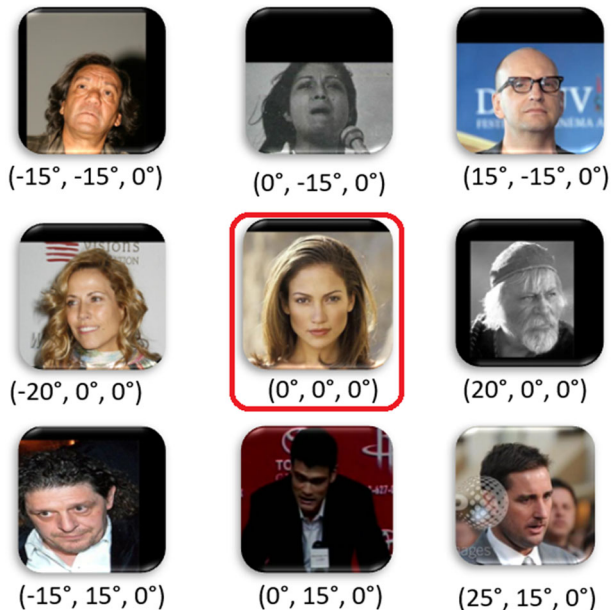


Fig. 3 Head poses close to the frontal pose

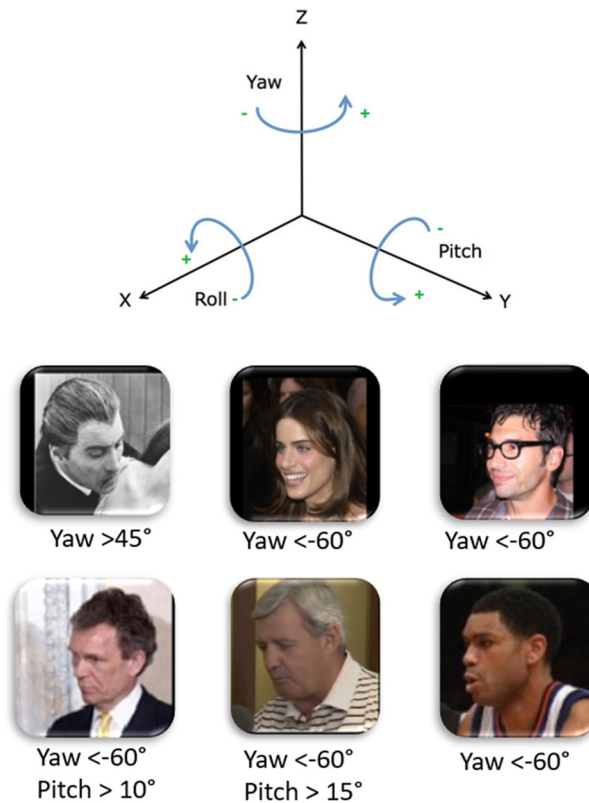


Fig. 4 Examples of large poses

Moreover, a face image captured under pose variation presents missing data, which can degrade the recognition rate.

Methods of face alignment are numerous and have shown impressive results with sophisticated techniques.

2D face alignment aims at establishing pose normalization if faces are in frontal or near-frontal poses as shown in Fig. 3. However, this transformation fails due to out-of-plane rotation. So, 2D face alignment has difficulties [41] when addressing large poses (Fig. 4). Yet, 3D face alignment consists in aligning faces despite the presence of out-of-plane rotations.

Whatever the method used for face alignment is, we must always take into account that the departure is based on facial landmarks.

The human face contains regions that make it unique even in the case of twins. These regions are called landmarks or/and keypoints (Table 3).

Landmarks: characteristic points in each face, such as the eyes, eyebrows, ears, chin, nose, mouth, etc. Their number is standard and fixed according to the applied algorithm. There are automatic algorithms of face annotation that generate landmarks. The use of landmarks serve to localize the salient regions of the face for face alignment, face morphing, face replacement, face recognition, etc.

Keypoints: characteristic points which characterize a single face. Indeed, two faces cannot contain the same keypoints, such as wrinkles, moles, warts, scars, etc.

Table 3 Summary of some face alignment methods based on fiducial landmarks

Author/Year	Main approach	Landmarks/keypoints	Use
Kazemi et al. [49]	Sparse subset of pixel intensities values extraction from input facial images. Facial landmarks estimation and regression using a set of regression trees.	194 landmarks	Face Alignment
Wenyan et al. [97]	Transform landmarks (from annotated datasets) to boundary heatmaps to aid feature learning.	landmarks extracted from boundary lines	Face Alignment
Dapogny et al. [20]	Chaining landmark localization tasks for predicting various numbers of landmarks.	Annotated datasets	Face Alignment
Paola et al. [5]	68 landmarks detection. Face sector identification based on a web-shaped model over the 68 landmarks.	according to sector coverage	Face Alignment
Abhinav et al. [53]	Ground-truth locations of 68 landmarks. Evaluate landmarks location and visibility. Occluded landmarks estimation.	68 landmarks	Face Alignment
Fard et al. [26]	Learning a smoother distribution of the facial landmark points. Guided network learning for original landmark points distribution.	Annotated datasets	Face Alignment Pose estimation

2.2.1 3D face alignment methods based on fitting 3D generic models to 2D faces

The human face is characterized by 68 landmarks which can provide information about the head pose. The fitting process consists in pasting a 3D face model to the 2D face using landmarks as references. This is performed by minimizing the difference between the face image and the 3D face model appearance. The purpose of fitting lies in the possibility of rotating the face and performing the alignment to a frontal pose. Fitting is a method used for 3D face alignment, especially in medium poses. However, in large poses, it is very challenging because of the dramatic appearance variations when getting closer to the profile view (Table 4).

In [100], the authors introduced a 3D Dense Face Alignment (3DDFA) which fits the 3D morphable model (3DMM) [12]. 3DDFA synthesizes face appearances by labeling invisible landmarks due to large poses. Its objective is to skip 2D landmark detection and start from 3DMM fitting. HPEN [101] aims at fitting the 3DMM to the 2D faces captured in the wild. The approximation method is also performed to avoid iterative visibility estimation of the masked landmarks in large poses. In addition, an identity-preserving normalization is carried out by correcting 3D transformation and anchoring adjustment in the meshed image. In the same context, a method proposed in [79] uses the Basel Face Model (BFM) [37] for 3D face alignment and keypoints locations. It consists of a deep evolutionary model integrating sparse 3D Diffusion Heap Maps (DHM) for pose assistance. CNN is used for feature extraction and Recurrent Neural Network (RNN) is utilized for learning.

The methods already quoted have achieved the best results in FR framework, including face alignment. However, the big challenge is always evoked when dealing with large poses. Their main drawback is therefore related to the limited geometry of the 3D models used. On the other hand, the use of a 3D model, such as 3DMM or BFM, to establish fitting always leaves a common signature in the extracted features.

Table 4 Summary of some 3D face alignment methods based on 3D morphable models fitting

Author/Year	Face alignment main approach	Datasets
Xiangyu et al. [101]	2D/3D landmarks marching 3DMM fitting Pose and expression normalization Possion Editing method for invisible region filling	Multi-PIE LFW
Xiangyu et al. [100]	3DMM fitting with cascaded CNN HOG features extraction combined with a linear regressor for landmarks refinement and invisible landmarks labeling	AFLW
Zhanfu et al. [2]	5 landmarks detection using MTCNN Facial pose estimation based on the correspondence between the 2D face and the 3DMM Adaptive template generation Face alignment based on the optimal reference template	IJB-A IJB-C CPLFW
Jiang et al. [42]	Spacial facial features extraction using Spacial Group-wise Feature enhancement 2D/3D landmarks estimation based on 3DMM and controlled by different loss functions	AFLW AFLW2000-3D
Jiwoo et al. [44]	3DMM fitting network based on a UV encoder and UV EB-GAN UV completion network for face alignment discrimination	300w-LP AFLW2000

2.2.2 3D face alignment methods based on 3D face model reconstruction

This process consists in reconstructing a 3D face model from a 2D face to have its own model for each input 2D face image without the need for a 3D model, such as 3DMM, BFM, or any external data (Table 5). Indeed, each 3D reconstructed model has its own characteristics and parameters. Thereafter, the 3D reconstructed model and the 2D landmarks are correlated by a specific technique.

DeepFace [82] modelizes a 3D face based on the extracted 67 fiducial points. Thus, this method consists in wrapping the detected facial crop to a 3D frontal model after mesh reconstruction by Delaunay triangulation. Also, the 67 anchor points are fitted to the obtained 3D shape to get correspondence between the 67 detected fiducial points and their 3D references. In the same context, another work used the Iterative Closest Points (ICP) [45] algorithm to perform correspondence between each reconstructed 3D face and the ground truth point cloud. Then, normalized mean error (NME) is calculated by the face bounding box size.

Feng et al. [28] proposed a new approach for 3D face reconstruction using UV space as a position map [85]. The UV position map represents the full 3D plot of facial structure with alignment information. It is a 2D image recording 3D positions of all the points in UV space. So, the full facial geometry is reconstructed along with the semantic meaning and it is regressed to get aligned faces.

Table 5 Summary of some 3D face alignment methods based on face model reconstruction from the input 2D face image

Author/Year	Main approaches of face alignment	Datasets
Taigman et al. [82]	3D face modeling using affine transformations based on 67 fiducial points. Face representation from a nine-layer deep neural network.	SFC LFW YTF
Feng et al. [28]	UV position map generation for a complete 3D shape reconstruction. A simple CNN to regress from a single 2D image.	Florence AFLW2000-3D AFLW-LFPA
Feng et al. [59]	3D shapes reconstruction based on 2D face image landmarks. Using reconstructed 3D shapes to refine landmarks based on cascaded landmark regressors and 3D shape regressors. Mapping 3D-to-2D landmarks for 3D face shape and landmarks correlation.	Multi-PIE AFLW AFLW2000-3D CFP
Tu et al. [87]	3D face reconstruction and alignment using 2D-Assisted Self supervised Learning (2DASL) method based on the 3DMM. 3DMM coefficient regression. 3D face model learning and better model reconstruction based on 2D landmarks.	AFLW2000-3D AFLW-LFPA
Browatzki et al. [13]	Face reconstruction based on low-dimensional face embedding. Color images generation to the prediction of landmark heatmaps.	300-W AFLW WFW
Zeyu et al. [70]	Self-aligned dual face regression network (SADRNet) for face pose estimation and face shape prediction. UV position map for correspondence between the face shape and the input 2D image.	AFLW2000-3D Florence

The previously cited works have used methods establishing alignment with face model reconstruction without 3D generic model basis, which is challenging but had good results. So, no 3D model shape or template restriction was present. In our method, reconstruction of 3D models for each 2D face is carried out as explained in the following section.

3 Proposed method

Conventional pipeline consists of face detection, face alignment to get frontal pose, face representation that has to be trained in the DCNN, and finally face classification to establish identification. Face detection and face alignment are preprocessing steps. In the figure below (Fig. 5), our global pipeline is presented.

A specification of the main algorithm of the proposed method is presented in Algorithm 1. The different steps are detailed in the following subsections.

```

Begin
Read input images from the dataset
for each image do
    2D_Face ← Face Detection and Cropping from input images (input image)
    2D_Landmarks ← Landmarks Detection and Extraction (2D_Face)
    3D_Face, 3D_Landmarks ← 3D Face Reconstruction (2D_Landmarks)
    Aligned_Face ← 3D Alignment (2DFace, 3DFace, 2D_Landmarks, 3D_Landmarks)
end for
Train_DCNN (DCNN)
Recognize faces and establish verification
End
    
```

Algorithm 1 Overall.

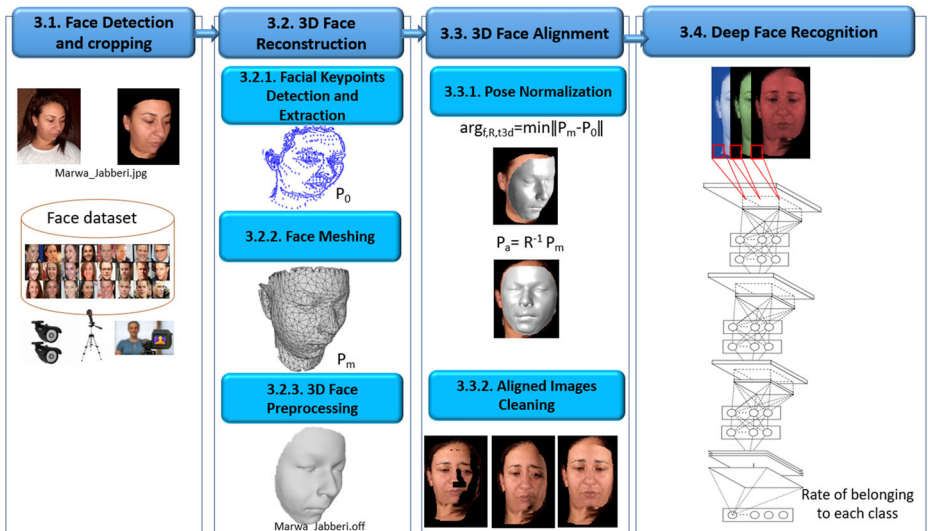


Fig. 5 Overview of the proposed method

3.1 Face detection and cropping

Before detecting faces in the images, we eliminate the duplicated images and check the labels. For face detection, Modified Viola–Jones algorithm [63] is used.

When it first appeared [92], this method was effective in the detection of faces in frontal position; however, following certain modifications, it has become sophisticated in all scenarios. So, the face, which is our region of interest (ROI), can be detected under various poses, various illumination conditions, different skin colors, and complex backgrounds while maintaining considerable speedup by parallelizing the training. Once the face is detected, bounding boxes are randomly generated around the detected window (Fig. 6).

When facial detection is established, all images are resized in the same scale. In case of images having multiple faces, each detected face is labeled manually and assigned to the appropriate class.

3.2 3D Face reconstruction

In this paper, we revisit the alignment step which consists in searching landmarks based on global shape or texture models to configure landmarks locations. However, under some view angles, landmarks are invisible. So, performance decreases for non-frontal faces and invisible landmarks are considered as self occlusions. It is for this reason that face reconstruction is required. The difference between using a 3D generic model and a 3D reconstructed model is that each 2D face has its own 3D model which preserves texture, shape, and other features. The use of a generic model, such as BFM or 3DMM, causes a common signature between all faces, which increases the error rate afterwards.

3D reconstruction is established by keypoints detection which is added to the traditional fiducial landmarks (Fig. 7). Indeed, the addition of supplementary keypoints to face features is helpful in the reconstruction stage because the 68 landmarks are not enough for 3D mesh creation.

3.2.1 Facial keypoints detection and extraction

First, we start by locating the 68 fiducial points using the facial landmark detector included in the dlib library and OpenCV presented in [71].

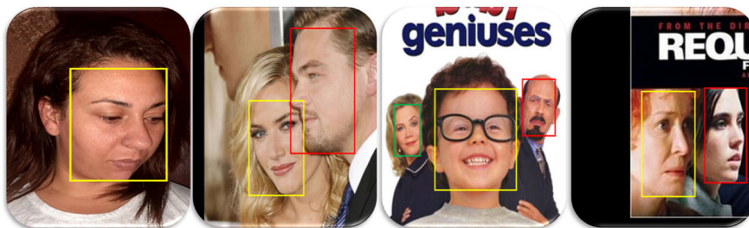


Fig. 6 Face detection using Modified Viola–Jones algorithm

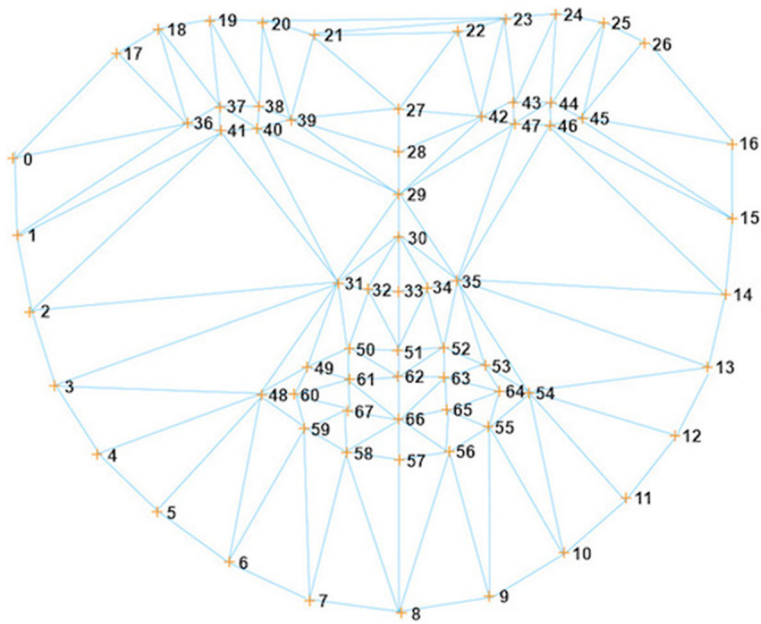


Fig. 7 68 traditional facial landmarks

Require: 2D_Face

Ensure: 2D_Landmarks

$68_{Landmarks} \leftarrow$ Estimate the location of 68 (x, y)-coordinates that map the facial structures on the face (2DFace) using dlib library

$Edges_{Canny} \leftarrow$ Edge Detection (2DFace, 'Canny')

$Edges_{Prewitt} \leftarrow$ Edge Detection (2DFace, 'Prewitt')

Regions \leftarrow MSER Features Detection (2DFace)

$2D_{keypoints} \leftarrow$ Reshape ($68_{Landmarks}$, $Edges_{Canny}$, $Edges_{Prewitt}$, Regions)

Algorithm 2 Keypoints detection and extraction.

The 68 (x, y) extracted landmarks allow to delineate the facial surface in the face image as shown in Fig. 8. Thus, our new ROI is delimited by the jaws and eyebrows keypoints. This method is tested under large poses and this step is successfully performed.

Our choice of the 68 facial landmarks detector was made following a series of tests and experiments that proved robustness against large poses. They are detailed in the self-evaluation section.

According to the state-of-the-art studies, the presence of out of plane or invisible landmarks is noted in large poses. So, keypoints are added since the 68 landmarks are not enough for 3D face reconstruction (Algorithm 2). Indeed, this is our basic contribution.

The edges in the face image are detected using Canny and Prewitt edge detection algorithms [91]. Only the features in the delimited ROI are kept.

The Canny method consists in finding edges by looking for local maxima in the image gradient. The edge function calculates the gradient using the derivative of a Gaussian filter.



Fig. 8 68 landmarks detection using dlib library: First row: input face images, Second row: numbered 68 landmarks

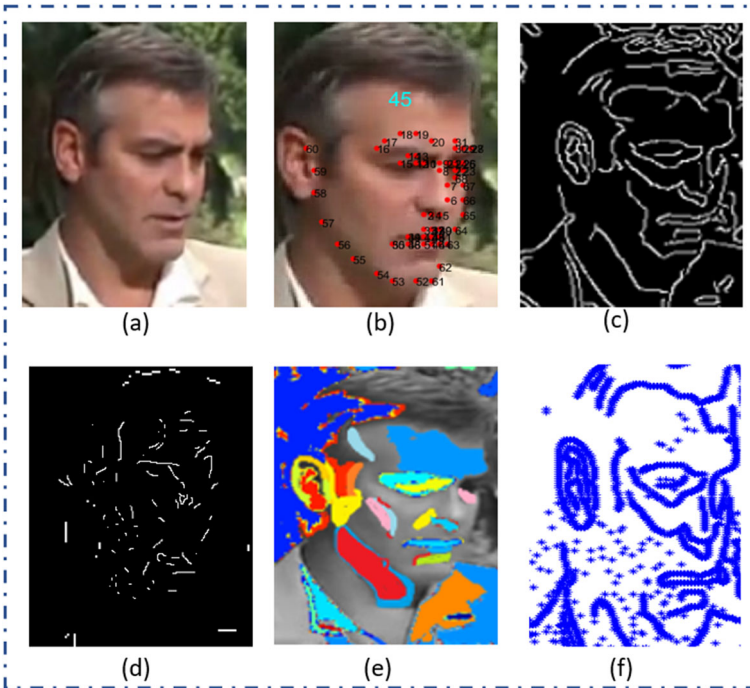


Fig. 9 Facial keypoints detection: (a) Input face image, (b) 68 Landmarks, (c) Detected edges using Canny, (d) Detected edges using Prewitt, (e) Detected regions using MSER, and (f) All detected keypoints

Table 6 Examples of the number of extracted keypoints of two query face images (the images are the faces of two celebrities coming from the datasets we test)

Example	Fiducial landmarks	Canny	Prewitt	MSER	Total
George_Clooney.jpg	68	711	409	319	1507
Aishwarya_Rai.jpg	68	603	271	374	1316

This method uses two thresholds to detect the strong and weak edges, including the weak edges in the output if they are connected to the strong ones. By using two thresholds, the Canny method helps to detect the true weak edges which can represent wrinkles in the face (Fig. 9(c)).

On the other hand, the Prewitt method aims at finding the edges at the points where the image gradient is maximum using the Prewitt approximation to the derivative (Fig. 9(d)).

Since the output is a binary image, pixels with 0 values are found and they are extracted to be added to the other keypoints. We notice that the number of keypoints is variable for each given face.

In addition to edge detection, Maximally Stable Extremal Regions (MSER) features [77] are added. Indeed, using this descriptor (Fig. 9(e)) allows to obtain good identification of significant image parts, usually combined with high repeatability under typical image distortions. It also allows to get highlighting boundaries of the ROI, which are maximally stable extremal regions. Moreover, MSER helps to find correspondences between the image elements from two images with different viewpoints.

For each input 2D image, the detected keypoints number is not the same (Fig. 9(f)). Once keypoints are detected, they are extracted and saved under the same label as the image in order to be used in the 3D reconstruction process. In Table 6, we present the number of extracted keypoints of two query face images.

In this work, we also add other keypoints to the traditional 68 landmarks. We believe that the face contains more points that characterizing it.

Using two examples, Table 6 shows that each face has a variable number of characterizing weights in each step of features extraction. The number of keypoints is useful in 3D face reconstruction, 3D face processing (mesh subdivision), face fitting, and face alignment process. The number of keypoints is required for face meshing.

3.2.2 Face meshing

Once the keypoints are extracted, we start meshing the ROI using Delaunay triangulation [11]. Algorithm 3 presents the main steps of 2D face meshing.

Require: $2D_{keypoints}$

Ensure: $3D_Face, 3D_{keypoints}$

$3D_{keypoints} \leftarrow$ Delaunay Triangulation ($2D_{keypoints}$)

$3D_Face \leftarrow$ 3D Mesh Generation ($3D_{keypoints}$)

Write off ($3D_Face$)

Algorithm 3 2D face meshing.

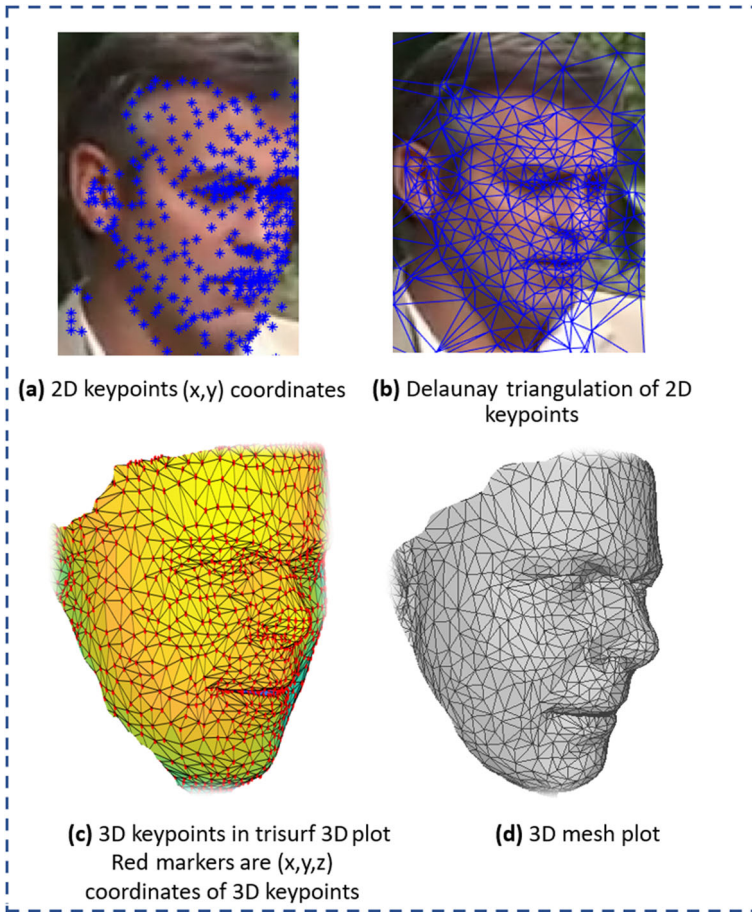


Fig. 10 3D face reconstruction from single 2D face image: Step by step image meshing (from 2D keypoints to 3D mesh)

Delaunay triangulation creates triangulations of a set of points and ensures that the circumcircle associated with each triangle contains no other points in its interior that depends on its neighborhood. Delaunay triangulation derived from the extracted facial keypoints is shown in Fig. 10.

After the triangulation process, we obtain facial points in 3D domain, derived from the facial keypoints in 2D domain using n , which is the number of extracted landmarks. It is worth noting that n is not the same for each given face (P_0 : Initial Points, P_m : Meshed Points).

$$P_0 = [x_1, y_1, x_2, y_2, \dots, x_n, y_n]^T \in \mathbb{R}^{2.n*1} \tag{1}$$

$$P_m = [x_1, y_1, z_1, x_2, y_2, z_2, \dots, x_n, y_n, z_n]^T \in \mathbb{R}^{3.n*1} \tag{2}$$

As previously mentioned, face cropping is performed to extract the face from the image, but we notice that a part of the background is still there. This part is useful in the alignment step; however, in the reconstruction of the 3D face, it should be ignored because we only

need the salient part of the face. If the background in the 3D reconstruction is left, this will be very demanding in terms of time and complexity.

3.2.3 3D face preprocessing

This step is very important since the obtained mesh is not in good quality due to several factors, such as mesh regularity and holes coming from self occlusions. Vertices with no connections can also be found. In Algorithm 4, we present the steps to be followed to perform 3D face preprocessing.

Require: $3D_{Face}$, $3D_{keypoints}$, $2D_{keypoints}$, $68_{landmarks}$

Ensure: $3D_{Processed\ Face}$, $3D_{keypoints}$

$3D_{patch\ Face}$, $3D_{keypoints} \leftarrow$ Region Growing ($3D_{Face}$, seed, radius, distance)

$Diagonal_{keypoints} \leftarrow$ Diagonal Keypoints Extraction ($3D_{patch\ Face}$, $3D_{keypoints}$, $2D_{keypoints}$, $68_{landmarks}$)

$3D_{update\ Face} \leftarrow$ 3D Mesh Update ($3D_{keypoints}$, $Diagonal_{keypoints}$)

$3D_{Processed\ Face} \leftarrow$ Butterfly Mesh Subdivision($3D_{update\ Face}$, iterations, edge threshold)

$3D_{Processed\ Face} \leftarrow$ Ball Pivoting Algorithm ($3D_{update\ Face}$, radius, angle)

Algorithm 4 3D face preprocessing.

First of all, we extract the facial surface using Region Growing [6], which is a segmentation algorithm suitable for 3D mesh. The nose tip is used as a seed point and several tests are performed to determine the extraction radius suitable for any face shape ($r = \text{length of Bounding Box} * 0.6$). Then, the geodesic distance is used to obtain an oval shape, as shown in Fig. 11. Indeed, the keypoints residing around the jaws and their neighborhoods are taken into consideration.

Once the suitable facial region (patch) is extracted from the initial generated mesh, we locate the diagonal of the face from the annotated landmarks (28, 29, 30, 31, 34, 52, 63, 67, 58, 9), as shown in Fig. 12(b). We also extract other facial diagonal keypoints having the same coordinates on the y axis as the last ones. Then, we start generating symmetrical vertices to the y axis of each facial landmark while considering x and z axes, as shown in

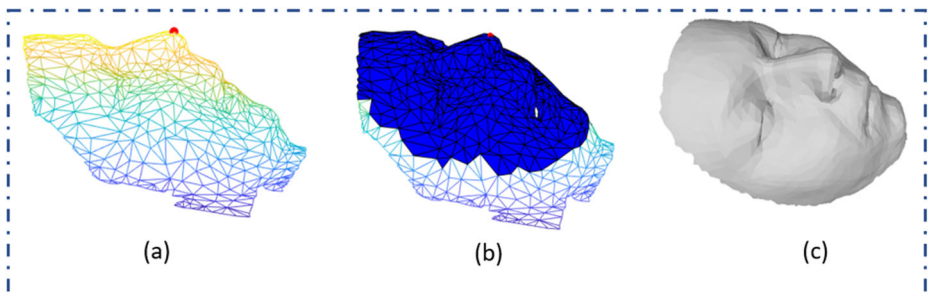


Fig. 11 3D facial surface extraction: (a) localization of the nose tip, (b) Facial patch detection, (c) Extracted face

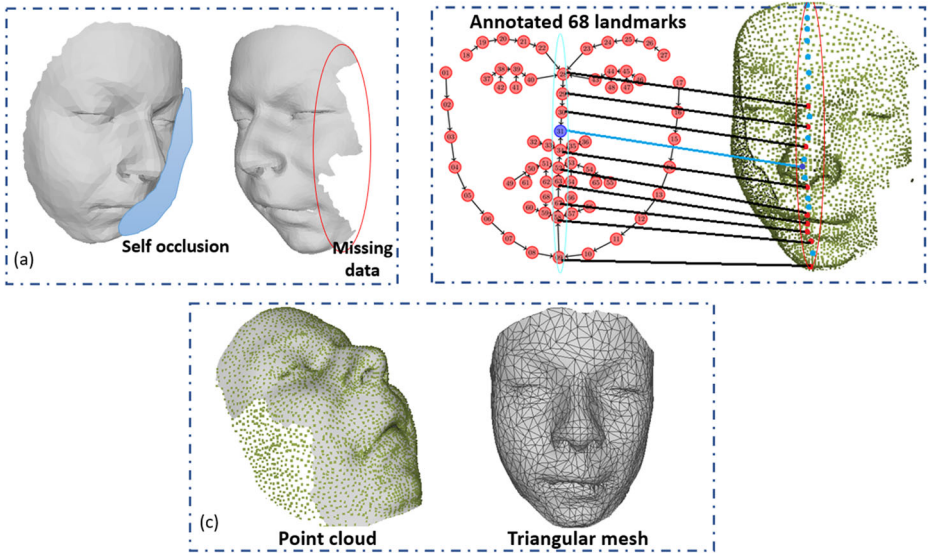


Fig. 12 3D mesh reconstruction (Update) by adding the missing parts: (a) Missing parts of the face caused by large poses, (b) Localization of facial diagonal keypoints referencing to the 68 annotated landmarks: The red points represent the number of landmarks: 28, 29, 30, 31, 34, 52, 63, 67, 58, and 9. The blue ones represent other detected keypoints having the same y coordinates, (c) Missing parts reconstruction

Fig. 12(c). This allows to solve the problem of missing parts or self occlusions caused by large poses and profile views (Fig. 12(a)).

After adding the missing parts of the 3D face, the quality of the preprocessed mesh in the context of good reconstruction is improved for the pose normalization task. Remeshing to connect the new vertices and the facial surface subdivisions of the mesh is performed using the Butterfly subdivision algorithm [60] and the Ball Pivoting Algorithm (BPA) [8] for triangular interpolation (Fig. 13) .

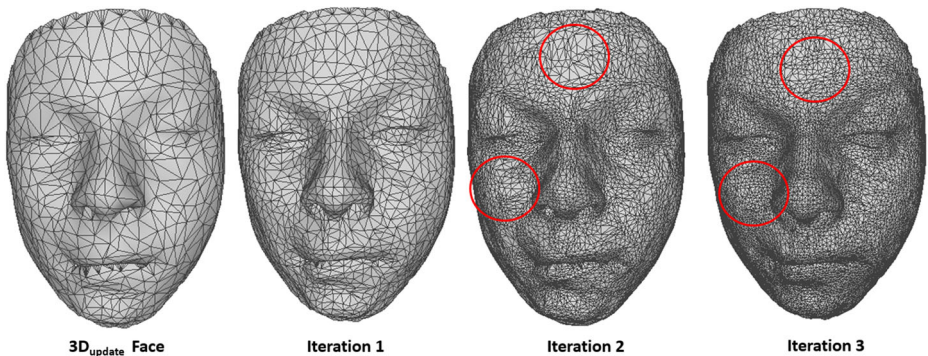


Fig. 13 Remeshing process: 3 mesh subdivision iterations using Butterfly algorithm. Red circles show interpolating triangulation using BPA

The Butterfly algorithm is used for mesh subdivisions and vertices connections. This process is very essential in 3D reconstruction to produce other vertices whose purpose is to achieve mesh regularity controlled by BPA to preserve the facial shape.

Using the butterfly algorithm, we normalize all the 3D reconstructed faces to a defined number of vertices and facets. Indeed, the original meshes do not have the same parameters since the number of extracted landmarks is variable from one face to another.

For each facet consisting of 3 vertices (3 coordinates in the 3D space x,y, and z), BPA pivots around an edge (which connects two vertices) until it touches another vertex, forming another triangle. So, BPA builds relationships between vertices having no connections, which improves the mesh regularity. This process is iterated until connecting all the vertices in the mesh.

BPA is a very used and efficient technique for mesh interpolation. It exhibits linear time complexity and robustness in the given 3D meshes. Although these two techniques are old, they are very efficient. In the experimental part, we justify our choice using some discriminating values.

3.3 3D face alignment

3.3.1 Pose normalization

After 3D mesh reconstruction and preprocessing, we wrap all the detected 2D facial keypoints by projecting the 3D reconstructed face onto the image plane using the Weak Perspective Projection [14], based on the 2D positions of the 3D points on the image plane. The following Algorithm summarizes the main steps.

Require: 2D_Face, 2D_keypoints, 3D_Processed Face, 3D_keypoints

Ensure: Aligned_Face

[f, R, t_{3d}] ← Project Vertex (2D_Face, 2D_keypoints, 3D_Processed Face, 3D_keypoints)

[f, R, t_{3d}, 3D_keypoints] ← Wrap Reconstructed 3D Model (2DFace, 2D_keypoints, 3D_Processed Face, 3D_keypoints)

Aligned_Face ← Pose Normalization (f, R, t_{3d}, 2DFace, 2D_keypoints, 3D_keypoints, 3D_Processed Face)

Aligned_Face ← Image Cleaning (Aligned_Face)

Algorithm 5 3D alignment.

Then, we fit the 3D obtained face by minimizing the difference between the 2D extracted landmarks and their references in our 3D reconstructed model while considering the rotation parameters (R is 3*3 matrix constructed with pitch (ϕ), yaw(γ), and roll(θ)), the translation vector t_{3d}, and the scale factor f given by the normalization process.

$$\text{arg}_{f,R,t_{3d}} = \min ||P_m - P_0|| \quad (3)$$

The rotation matrix is obtained by multiplying the following three matrices:

$$R_x(\phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{pmatrix} \quad (4)$$

$$R_y(\theta) = \begin{pmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{pmatrix} \quad (5)$$

$$R_z(\gamma) = \begin{pmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (6)$$

In Fig. 14, we present the results of the fitting process when using our 3D reconstructed models. The first line presents celebrities faces taken from the datasets we test, the second line contains the fitting results.

The salient surface of the face is completely and perfectly wrapped by our reconstructed 3D model. The advantage of 3D reconstruction is that each identity has a specific 3D model, which is useful for alignment. This makes it unique and original. In fact, there is no common factor between the different identities. Indeed, this is useful for the recognition task.

Later, we perform pose correction for the alignment step. So, the 3D face designed by P_m in (7) is rotated by normalizing with R^{-1} to the frontal pose with 0° view centered by the nose tip and considering the pose map of the 2D extracted keypoints. This step is iterated until the face is aligned (P_a) to the desired view according to the pitch (θ), yaw (γ), and roll (ϕ) values of the frontal pose.

$$P_a = R^{-1}P_m \quad (7)$$

Once the 3D face is normalized to the frontal pose, correspondence between 3D and 2D keypoints is redone to refine the new 2D keypoints location.

Following a bibliographic study we performed, we notice that face alignment methods using generic 3D models have a problem of breaking correspondence, especially in cases of large poses. Indeed, the keypoints on the face contour boundary are not consistent. In



Fig. 14 Fitting results using our 3D reconstructed models: The first line includes original face images, the second line contains fitting results

addition, the shape of the 3D generic model is always existent. This implies that after the fitting process, all the faces will have a common touch despite the different identities, simply because they are fitted with the same 3D generic model. For this reason, full reconstruction of the 3D face of each given 2D face is efficient and recommended. So, each 2D face will have its own 3D modeling which makes it truly original following the fitting and the alignment steps.

3.3.2 Aligned image cleaning

After the fitting and the alignment processes, we notice that the images obtained are not in good condition and they contain holes and missing parts due to alignment.

Some preprocessing operations are performed to clean the resulting images and to increase the recognition rate. It is not possible to generate a reasoned face image just like the one taken in the frontal view. So, artifacts are treated using the mirroring method [22], whose purpose is to fill the holes and the missing parts caused by alignment.

In Fig. 15, the graphical results of 3D face alignment when applying our method are presented. The blue circles show our method robustness and justify our contribution at the level



Fig. 15 Graphical results of face cleaning and alignment: (a) Input image, (b) 3D Face alignment, (c) Results of image cleaning and alignment

of keypoints addition, which serves to detect more regions and to wrap all the visible parts of the face. In fact, more keypoints extraction involves good 3D face reconstruction, which leads to fit the whole face region to get better face alignment. The purpose of such face alignment is to increase face recognition results, no matter how challenging the conditions are. In This work [39], another use cases of our alignment approach is presented.

3.4 Deep face recognition

After face frontalization and preprocessing, we move to face verification using DCNNs which eliminate the need for manual features extraction. Otherwise, the features are learned directly. We train our DCNN on a multi-class face dataset. To establish this operation, our main objectives are fast GPU-implementation of a DCNN to win a face image recognition contest and to search for successful DCNN applications for such big datasets. Applying DCNN to aligned facial images makes the network more robust to small registration errors.

In our work, we tried several DCNNs and we kept the best recognition rates obtained for each dataset. Our DCNN is therefore trained on an aligned RGB face image. The image size is adapted to the input layer of each tested DCNN.

Our input consists of an RGB image of the aligned face which is given to a convolutional layer (CL) and resized according to the CL characteristic of each tested DCNN. Indeed, AlexNet [52] has given the best recognition rates that will be detailed in the experimental part of this paper.

4 Experimental results

Using our method, we present the experiments conducted on YTF, LFW, and BIWI datasets, which are well-known benchmarks for face recognition. Our implementation is based on the dlib library using Python 3, MatConvNet, Image Processing, and Graph MATLAB toolbox for 3D mesh processing. Indeed, MeshLab linked to the NVIDIA packages is used to accelerate training. All our experiments were carried out using NVIDIA CUDA development 9.2 and were run on intel (R) Core (TM) i7-7500U, 2.70 GHz and 2.90 GHz with 8GO RAM.

4.1 Experimentation and results on LFW dataset

Labeled Faces in the Wild is a big dataset for face verification testing in unconstrained domains (lighting, poses, facial expressions). It contains 13,233 face images of 5,749 different identities collected from the web. It includes 1,680 people having two or more images against 4,069 people having only single image in the dataset.

In our experiments, we used the configuration described in paper [36] related to the dataset, and we only used the LFW samples. No outside data were used. Two protocols are presented in LFW dataset: image-restricted and image-unrestricted protocols.

The restricted protocol has image-restricted settings: binary labels are available. So, “matched” or “mismatched” verification for pairs of images is performed. On the other hand, the unrestricted protocol has unrestricted setting: identity information of the person in the image is available which helps to make new pairs of images.

Following this experimentation, we tested several DCNNs and the best recognition results are obtained using AlexNet. They were **98.37%** with the restricted protocol and **97.28%** when using the unrestricted protocol. Table 7 presents a comparison between our

Table 7 Comparison of FR rates with some existing methods on LFW dataset

Method	Restricted	Unrestricted
Shi et al. [76]	52.53%	
Kuang et al. [17]	88.27%	
HPEN [101]	92.80%	95.25%
DeepFace [82]	97.15%	97.35%
Our Method	98.37%	97.28%

results and those of the existing methods using different alignment methods, as described in the previous sections.

4.2 Experimentation and results on YTF dataset

YouTube Faces dataset [96] includes 3,425 YouTube videos of 1,595 different subjects. The used classes are the same as in LFW (a subset of celebrities presented in the LFW dataset [36]). The videos were taken by professional photographers and were divided into 5,000 video pairs and 10 splits. They were used to evaluate the video-level face verification. The images of this dataset are not in good quality due to acquisition problems. So, a preprocessing step, including smoothing and other filters was primordial.

In this paper, we performed our experiments employing the restricted protocol, which limits the information available for training to the same/not-same labels in the training splits.

Before performing 3D alignment, FR was tested via different DCNNs to check if alignment increases the recognition rate. Using AlexNet, the recognition rate was **99.14%**. In Table 8, a comparison with some related works is presented.

4.3 Experimentation and results on Biwi dataset

The BIWI dataset includes 15,678 frames collected from videos of 20 individuals: 6 women and 14 men (there are ones who were recorded twice). There are 24 sequences acquired with a Kinect sensor and collected in under-controlled conditions and different head poses.

In our experimentation, we used 2D frame images (RGB) presented in the dataset. We performed the same processing steps used for the two other tested datasets. Then, we applied our proposed method of 3D face alignment and pose normalization.

Table 8 Comparison with the state-of-the-art on YTF dataset

Method	Accuracy
Aligned faces provided by YTF	78.19%
Deepface [82]	91.4%
CosFace [94]	97.4%
SphereFace [58]	95%
RegularFace [99]	96.7%
Fenggao et al. [84]	94.6%
Rao et al. [68]	94.28%
Our method	99.14%

Table 9 Comparison with the state-of-the-art on BIWI dataset

Method	Accuracy
PIFS [10]	96.99%
Fei Gao et al. [29]	97.50%
Our method	97.92%

For FR, we followed the experimental protocol used by several works in the literature for this dataset. So, we randomly split the dataset into 70% for training and 30% for testing and verification. Using AlexNet, the recognition rate was **97.92%**. In Table 8, a comparison with some related works is presented (Table 9).

4.4 Self evaluation

We carried out a series of tests to justify our qualitative and quantitative choices of the different parameters and techniques. Apart from highlighting the robustness of our contribution through the rates obtained, we would also like to emphasize the quality of our work.

First of all, we start by justifying the use of the 68 landmark detector. As it was mentioned in the proposed method, we used dlib and OpenCV through Python3. This technique gave the best results for face annotation compared to Chow-Liu algorithm [18], which is widely used in recent face landmarks detection methods although it is an old technique (Fig. 16(b)), and compared to the Gaussian-Newton method [89], which is also widely used in face alignment (Fig. 16(c)). Comparison can also be made through the graphic results in Fig. 16.

The used technique established landmarks detection in almost all pose variations. Contrarily to some other techniques, we obtained errors of landmarks detection in critical scenarios or bad locations that would be disturbing during mesh reconstruction.

Our first contribution consists in adding more keypoints to the traditional 68 facial landmarks. This is useful for 3D model reconstruction which is used in the alignment process.

So, is 3D reconstruction perfect?

To answer this question, an experiment was carried out. An evaluation of 3D reconstruction was made based on the BU3DFE dataset [83], which contains 3D meshes accompanied by 2D images just to make sure that our reconstruction is perfect and close to the 3D faces as taken by 3D acquisition devices.

We used Mean Absolute Error (MAE) evaluation metric which measures the average magnitude of errors between prediction (3D reconstructed faces) and real 3D faces.

The average MAE of 3D reconstructed faces decreases with each addition of other keypoints (Fig. 17). This justifies the addition of other points to accomplish this task of reconstruction. However, the rates obtained are not within the standards.

For this reason, 3D mesh preprocessing was performed to conduct mesh regularization and to further decrease MAE, which would guarantee the alignment phase, as shown in Fig. 18.

In Fig. 19, the histogram presents a quantitative study of the number of vertices and facets during the 3D reconstruction phase. We perform 3 iterations of mesh subdivision using Butterfly algorithm in the remeshing step. This choice is established after a series of tests. For interpolating triangulation using BPA, pivoting ball radius is $=3.3231$ and the angle threshold is $=90^\circ$.



Fig. 16 68 facial landmarks detection: (a) Input face image, (b) 68 landmarks detection using Chow-Liu algorithm, (c) 68 landmarks detection using the Gaussian-Newton method, (d) 68 landmarks detection using dlib and OpenCV

Once 3D model reconstruction is performed for each given 2D face, fitting to wrap all the detected 2D facial landmarks is conducted by projecting the 3D reconstructed faces onto the 2D ones.

As self-evaluation, the fitting process was tested using two widely-used existing models in face alignment, in addition to the model we generated. So, we noticed that the alignment process with fitting BFM (Fig. 20(a)) is not well-adapted to the 2D face due to projection

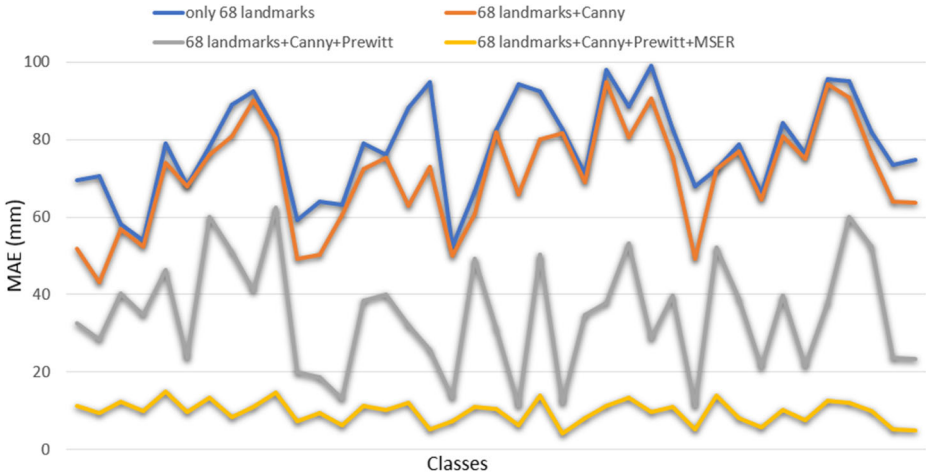


Fig. 17 MAE of the proposed 3D reconstruction method on BU3DF

errors. The shift is very remarkable. Apart from the cases of large poses, many images are missed because projection is unreachable.

When using 3DMM (Fig. 20(b)), the fitting process was successful under wide poses. We also noticed that facial expressions are well-illustrated on the obtained model. This is due to the reconstruction of this generic model learned from 10,000 faces in the wild. However, using this model has one drawback consisting in image meshing each time the shape of the 3DMM is present in all faces. This implies that all the identities have the same signature, which degrades face frontalization.

Performing fitting with an appropriate 3D face model, as shown in Fig. 20(c), aids in preserving identity at the level of pose correction. All the 2D keypoints undergo this change of plane while referencing to the 3D ones.

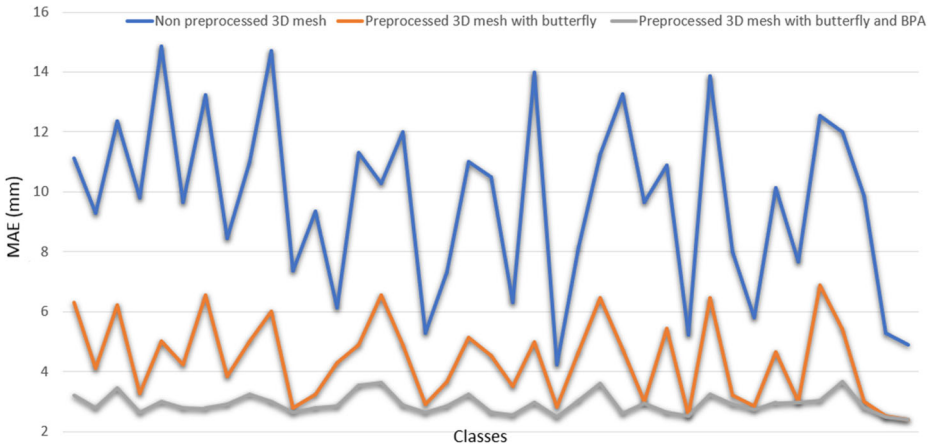


Fig. 18 MAE of the proposed processed 3D reconstruction method on BU3DF using butterfly and BPA algorithms

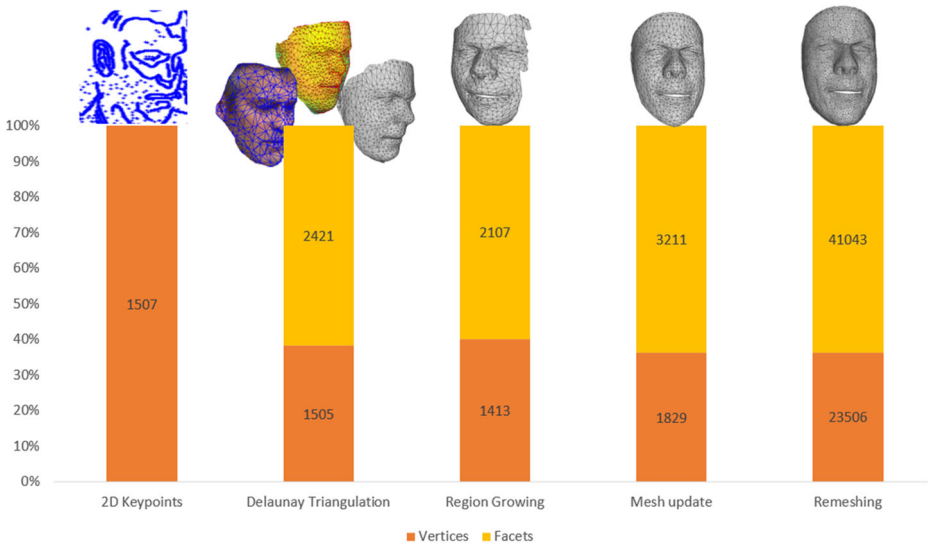


Fig. 19 Number of vertices and facets during the 3D reconstruction process of the query face image

Moreover, quantitative tests were carried out to justify and highlight our contribution. A recognition test was therefore established after having carried out the alignment process using the previously mentioned fitting methods. We used the same technique of keypoints



Fig. 20 Fitting results: (a) Fitting process using BFM, (b) Fitting process using 3DMM, (c) Fitting process using our reconstructed model

Table 10 Face recognition rates of aligned faces on YTF dataset

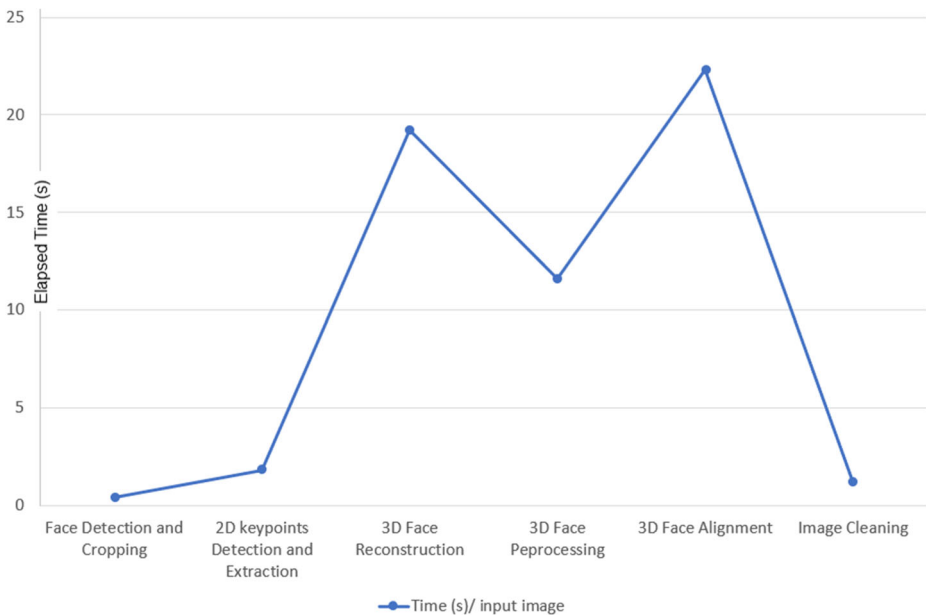
Method	Accuracy
Alignment using BFM	73.68%
Alignment using 3DMM	92.03%
Our method	99.14%

Table 11 Face recognition rates of aligned faces on LFW dataset, testing restricted and unrestricted protocols

Method	Restricted	Unrestricted
Alignment using BFM	81.97%	80.6%
Alignment using 3DMM	74.64%	79.15%
Our method	98.37%	97.28%

Table 12 Face recognition rates of aligned faces on BIWI dataset

Method	Accuracy
Alignment using BFM	80.21.68%
Alignment using 3DMM	91.15%
Our method	97.92%

**Fig. 21** Consumed time computing during the preprocessing step of FR pipeline per a query image

projection and keypoints marching. The recognition rates are presented in Tables 10, 11, and 12.

Indeed, BFM and 3DMM are two different generic models used in the fitting process. Yet, for pose normalization, both image cleaning and image classification were carried out in the same way to be able to establish comparisons between results.

To ensure that our approach is efficient and effective, the time factor is considered. The following curve shows the time consumed in each step (Fig. 21).

4.5 Discussion

Our contribution consists in applying 3D face alignment to FR. The results obtained are among the best ones thanks essentially to the efficiency of our 3D face alignment method.

Adding keypoints consists in covering the cropped facial surface, which reduces the number and the size of regions hidden by poses. This guarantees a sophisticated 3D mesh reconstruction from a single input face image. The aim of 3D reconstruction is to wrap maximum keypoints when the fitting process is established. This process facilitates face rotation with a slight damage to the 2D face image.

5 Conclusion

This paper presents a research on face recognition using DCNNs with appropriate training. We added keypoints to the 68 traditional fiducial landmarks using MSER, Canny, and Prewitt techniques.

We reconstructed 3D meshes based on Delaunay triangulation, followed by facial surface extraction using Region Growing algorithm, mesh subdivision, and remeshing using Butterfly and BPA algorithms.

Then, we projected the obtained 3D mesh onto the 2D image plane and wrapped it. This step was followed by pose correction whose purpose was to establish face alignment.

The recognition rates we found are justified by several factors, including the well-developed preprocessing steps and the efficient addition of more keypoints. This proves that 3D mesh reconstruction was conducted very carefully. So, the resulting images of faces were directly given to DCNNs without any intervention.

The results obtained are comparable to those reported in the state-of-the-art. In the near future, we are preparing other experiments on other existing benchmarks, such as LFPW and WLFW, using our proposed method.

Acknowledgements The research leading to these results has received funding from the Ministry of Higher Education and Scientific Research of Tunisia under the grant agreement number LR11ES48.

Funding The research work has funded from the Tunisian Ministry of Higher Education and Scientific Research under the grant agreement number LR11ES48.

Declarations

Financial interests The authors declare they have no financial interests.

Conflict of Interests The authors declare that they have no conflict of interest.

References

1. Ali W, Tian W, Din SU, Iradukunda D, Khan AA (2021) Classical and modern face recognition approaches: a complete review. *Multimed Tools Appl* 80:4825–4880
2. An Z, Deng W, Hu J, Zhong Y, Zhao Y (2019) APA: adaptive Pose alignment for pose-invariant face recognition. *IEEE Access* 7:14653–14670
3. Anwarul S, Dahiya S (2020) A comprehensive review on face recognition methods and factors affecting facial recognition accuracy. *Proc ICRIC 2019*:495–514
4. Arigbabu OA, Ahmad SMS, Adnan WAW et al (2017) Soft biometrics: gender recognition from unconstrained face images using local feature descriptor. [arXiv:1702.02537](https://arxiv.org/abs/1702.02537)
5. Barra P, Barra S, Bisogni C, De Marsico M, Nappi M (2020) Web-shaped model for head pose estimation: An approach for best exemplar selection. *IEEE Trans Image Process* 29:5457–5468
6. Beksi WJ, Papanikolopoulos N (2016) 3D region segmentation using topological persistence. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 1079–1084
7. Benmohamed A, Neji M, Ramdani M, Wali A, Alimi AM (2015) Feast: face and emotion analysis system for smart tablets. *Multimed Tools Appl* 74(21):9297–9322
8. Bernardini F, Mittleman J, Rushmeier H, Silva C, Taubin G (1999) The ball-pivoting algorithm for surface reconstruction. *IEEE Trans Vis Comput Graph* 5(4):349–359
9. Bhople AR, Prakash S (2021) Learning similarity and dissimilarity in 3D faces with triplet network. *Multimed Tools Appl* 80(28–29):35973–35991
10. Bisogni C, Nappi M, Pero C, Ricciardi S (2021) PIFS scheme for head pose Estimation aimed at faster face recognition. *IEEE Trans Biom Behav Identity Sci* 4(2):173–184
11. Boissonnat JD, Dyer R, Ghosh A (2018) Delaunay triangulation of manifolds. *Found Comput Math* 18(2):399–431
12. Booth J, Roussos A, Zafeiriou S, Ponniah A, Dunaway D (2016) A 3d morphable model learnt from 10,000 faces. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5543–5552
13. Browatzki B, Wallraven C (2020) 3fabrec: fast few-shot face alignment by reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 6110–6120
14. Bruckstein AM, Holt RJ, Huang TS, Netravali AN (1999) Optimum fiducials under weak perspective projection. *Int J Comput Vis* 35(3):223–244
15. Cao J, Hu Y, Zhang H, He R, Sun Z (2018) Learning a high fidelity pose invariant model for high-resolution face frontalization. In: Advances in neural information processing systems, vol 31
16. Chandrakala M, Durga Devi P (2021) Face Recognition using cascading of HOG and LBP feature extraction International conference on soft computing and signal processing. Springer, Singapore
17. Chou KP, Prasad M, Yang J, Su S-Y, Tao X, Saxena A, Lin W-C, Lin C-T (2021) A robust real-time facial alignment system with facial landmarks detection and rectification for multimedia applications. *Multimedia Tools Appl* 80.11:16635–16657
18. Chow CKCN, Liu C (1968) Approximating discrete probability distributions with dependence trees. *IEEE Trans Inf Theory* 14(3):462–467
19. Cui J, Zhang H, Han H, Shan S, Chen X (2018) Improving 2D face recognition via discriminative face depth estimation. In: 2018 International conference on biometrics (ICB), IEEE, pp 140–147
20. Dapogny A, Bailly K, Cord M (2019) Decafa: deep convolutional cascade for face alignment in the wild. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 6893–6901
21. Deeba F, Ahmed A, Memon H et al (2019) LBPH-Based enhanced real-time face recognition. *Int J Adv Comput Sci Appl* 10(5):274–280
22. Ding L, Ding X, Fang C (2012) Continuous pose normalization for pose-robust face recognition. *IEEE Sig Process Lett* 19(11):721–724
23. Dkhil MB, Wali A, Alimi AM (2018) Towards a new system for drowsiness detection based on eye blinking and head posture estimation. [arXiv:1806.00360](https://arxiv.org/abs/1806.00360)
24. Elaggoune Hocine et al (2021) Hybrid Descriptor and Patches Optimization for Face Recognition. In: 2021 1st International Conference On Cyber Management And Engineering (CyMaEn), IEEE
25. Fanelli G, Dantone M, Gall J, Fossati A, Van Gool L (2013) Random forests for real time 3d face analysis. *Int J Comput Vis* 101.3:437–458
26. Fard AP, Abdollahi H, Mahoor M (2021) ASMNet: a lightweight deep neural network for face alignment and pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 1521–1530
27. Fdhila R, Ouarda W, Alimi AM, Abraham A (2016) A new scheme for face recognition system using a new 2-level parallelized hierarchical multi objective particle swarm optimization algorithm. *J Inf Assur Secur* 11.6:385–394

28. Feng Y, Wu F, Shao X, Wang Y, Zhou X (2018) Joint 3d face reconstruction and dense alignment with position map regression network. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 534–551
29. Gao F, Li S, Lu S (2021) How frontal is a face? Quantitative estimation of face pose based on CNN and geometric projection. *Neural Comput & Applic* 33.8:3035–3051
30. Guides FR Articles, FindBiometrics. com.
31. Gupta S, Thakur K, Kumar M (2021) 2D-human face recognition using SIFT and SURF descriptors of face's feature regions. *Vis Comput* 37:447–456
32. He X, Niyogi P (2003) Locality preserving projections. *Adv Neural Inf Process Syst* 16
33. He X et al (2005) Face recognition using laplacianfaces. *IEEE Trans Pattern Anal Mach Intell* 27.3:328–340
34. Heseltine Thomas, Pears Nick, Austin Jim (2004) Three-dimensional face recognition: A fishersurface approach International Conference Image Analysis and Recognition. Springer, Berlin, Heidelberg
35. Huang Y-S, Chen S-Y (2015) A geometrical-model-based face recognition. In: 2015 IEEE International Conference on Image Processing (ICIP), IEEE
36. Huang GB, Learned-Miller E (2014) Labeled faces in the wild: updates and new reporting procedures. *Dept Comput Sci, Univ Massachusetts Amherst, MA, USA, Tech. Rep, vol 14, no 003*
37. Huber P, Hu G, Tena R, Mortazavian P, Koppen P, Christmas WJ, Kittler J (2016) A multiresolution 3d morphable face model and fitting framework. In: International conference on computer vision theory and applications. Vol. 5. SciTePress
38. Islem J, Wael O, Alimi AM (2017) Deep neural network features for horses identity recognition using multiview horses' face pattern. In: Ninth international conference on machine vision, (ICMV 2016). Vol. 10341. SPIE
39. Jabberi M, Wali A, Alimi AM (2023) Generative data augmentation applied to face recognition. In: 2023 International Conference on Information Networking (ICOIN). IEEE, pp 242–247
40. Jarraya I, Ouarda W, BS F, Alimi AM (2021) Sparse Neural Network for horse face detection in a Smart Riding Club Biometric System
41. Jeni LA, Tulyakov S, Yin L, Sebe N, Cohn JF (2016) The first 3d face alignment in the wild (3dfaw) challenge. In: Computer Vision-ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8–10 and 15–16, 2016, Proceedings, Part II 14. Springer International Publishing, pp 511–520
42. Jiang L, Xiao-Jun W, Kittler J (2019) Dual attention MobDenseNet (DAMDNet) for robust 3D face alignment. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops
43. Jing X-Y, Wong H-S, Zhang D (2006) Face recognition based on 2D Fisherface approach. *Pattern Recognit* 39(4):707–710
44. Kang J, Lee S, Lee S (2021) Competitive learning of facial fitting and synthesis using uv energy. *IEEE Trans Syst Man Cybern Syst*
45. Kapoutsis CA, Vavoulidis CP, Pitas I (1999) Morphological iterative closest point algorithm. *IEEE Trans Image Process* 8(11):1644–1646
46. Kar A, Neogi PPG (2020) Triangular coil pattern of local radius of gyration face for heterogeneous face recognition. *Appl Intell* 50(3):698–716
47. Karanwal S (2022) Robust local binary pattern for face recognition in different challenges. *Multimedia Tools Appl* 1-17
48. Kas M, El-merabet Y, Ruichek Y, Messoussi R (2020) A comprehensive comparative study of handcrafted methods for face recognition LBP-like and non LBP operators. *Multimed Tools Appl* 79:375–413
49. Kazemi V, Sullivan J (2014) One millisecond face alignment with an ensemble of regression trees. In: Proceedings of the IEEE conference on computer vision and pattern recognition
50. Khan M, Chakraborty S, Astya R, Khepra S (2019) Face Detection and Recognition Using OpenCV. In: 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), IEEE, pp 116–119
51. Kortli Y, Jridi M, Al Falou A, Atri M (2020) Face recognition systems: a survey. *Sensors* 20(2):342
52. Krizhevsky A, Sutskever I, Hinton GE (2017) Imagenet classification with deep convolutional neural networks. *Communications of the ACM* 60(6):84–90
53. Kumar A, Marks TK, Mou W, Ye W, Jones M, Cherian A, Koike-Akino T, Liu X, Feng C (2020) LUVLi Face alignment: Estimating Landmarks' Location, Uncertainty, and Visibility Likelihood. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition
54. Li D, Chen X, Becchi M, Zong Z (2016) Evaluating the energy efficiency of deep convolutional neural networks on CPUs and GPUs. In: 2016 IEEE international conferences on big data and cloud computing (BDCloud), social computing and networking (SocialCom), sustainable computing and communications (SustainCom)(BDCloud-SocialCom-SustainCom), IEEE, pp 477-484

55. Li Y et al (2021) Research On face recognition algorithm based on HOG feature. *Journal of physics: conference series*. Vol. 1757. No. 1. IOP Publishing
56. Ling H, Wu J, Huang J, Chen J, Li P (2020) Attention-based convolutional neural network for deep face recognition. *Multimed Tools Appl* 79(9):5595–5616
57. Liu Y, Schmidt KL, Cohn JF, Mitra S (2003) Facial asymmetry quantification for expression invariant human identification. *Comput Vis Image Underst* 91(1-2):138–159
58. Liu W, Wen Y, Yu Z, Li M, Raj B, Song L (2017) SpheroFace: Deep hypersphere embedding for face recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 212–220
59. Liu F, Zhao Q, Liu X, Zengal D (2018) Joint face alignment and 3D face reconstruction with application to face recognition. *IEEE Trans Pattern Anal Mach Intell* 42.3:664–678
60. Mandal C, Qin H, Vemuri BC (2000) Dynamic modeling of butterfly subdivision surfaces. *IEEE Trans Vis Comput Graph* 6(3):265–287
61. Moujahid A, Dornaika F, Arganda-Carreras I, Reta J (2021) Efficient and compact face descriptor for driver drowsiness detection. *Expert Syst Appl* 168(114334)
62. Muqet MA, Holambe RS (2019) Local binary patterns based on directional wavelet transform for expression and pose-invariant face recognition. *Appl Comput Inform* 15.2:163-171
63. Mutneja V, Singh S (2019) Modified Viola–Jones algorithm with GPU accelerated training and parallelized skin color filtering-based face detection. *J Real-Time Image Proc* 16(5):1573–1593
64. Napoléon T, Alfalou A (2014) Local binary patterns preprocessing for face identification/verification using the VanderLugt correlator. In: *Optical Pattern Recognition XXV*, vol. 9094. SPIE
65. Ning X, Nan F, Xu S, Yu L, Zhang L (2020) Multi-view frontal face image generation: a survey. *Concurr Comput Pract Experience* e6147
66. Oloyede MO, Hancke GP, Myburgh HC (2020) A review on face recognition systems: recent approaches and challenges. *Multimed Tools Appl* 79(37):27891–27922
67. Pandya JM, Rathod D, Jadav JJ (2013) A survey of face recognition approach. *Int J Eng Res Appl (IJERA)* 3(1):632–635
68. Rao Y, Lin J, Lu J, Zhou J (2017) Learning discriminative aggregation network for video-based face recognition. In: *Proceedings of the IEEE international conference on computer vision*, pp 3781–3790
69. Rao KS, Rajagopalan AN (2005) A probabilistic fusion methodology for face recognition. *EURASIP J Adv Signal Process* 2005.17:1–16
70. Ruan Z, Zou C, Wu L, Wu G (2021) Sadrnet: self-aligned dual face regression networks for robust 3d dense face alignment and reconstruction. *IEEE Trans Image Process* 30:5793–5806
71. Sagonas C, Antonakos E, Tzimiropoulos G, Zafeiriou S, Pantic M (2016) 300 Faces in-the-wild challenge: Database and results. *Image Vision Comput* 47:3–18
72. Shah JH, Sharif M, Raza M, Azeem A (2013) A survey: linear and nonlinear PCA based face recognition techniques. *Int Arab J Inf Technol* 10(6):536–545
73. Shan Shiguang et al (2002) Extended Fisherface for face recognition from a single example image per person. In: *2002 IEEE International Symposium on Circuits and Systems. Proceedings (Cat. No. 02CH37353)*, vol 2. IEEE
74. Sharif M, Naz F, Yasmin M, Shahid MA, Rehman A (2017) Face recognition: a survey. *J Eng Sci Technol Rev* 10(2)
75. Sharkas M, Elenien MA (2008) Eigenfaces vs. fisherfaces vs. ICA for face recognition; a comparative study. In: *2008 9th International Conference on Signal Processing, IEEE*
76. Shi L, Song X, Zhang T, Zhu Y (2019) Histogram-based CRC for 3D-aided pose-invariant face recognition. *Sensors* 19(4):759
77. Śluzek A (2016) Improving performances of MSER features in matching and retrieval tasks *European Conference on Computer Vision*. Springer, Cham, pp 759–770
78. Soltanpour S, Boufama B, Wu QJ (2017) A survey of local feature methods for 3D face recognition. *Pattern Recogn* 72:391–406
79. Sun B, Shao M, Xia S, Fu Y (2019) Real-time memory efficient large-pose face alignment via deep evolutionary network. [arXiv:1910.11818](https://arxiv.org/abs/1910.11818)
80. Sun H, Zhen X, Zheng Y, Yang G, Yin Y, Li S (2017) Learning deep match kernels for image-set classification. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp 3307–3316
81. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Rabinovich A (2015) Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 1–9
82. Taigman Y, Yang M, Ranzato MA, Wolf L (2014) DeepFace: closing the gap to human-level performance in face verification. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 1701–1708

83. Tang H, Huang TS (2008) 3D facial expression recognition based on automatically selected features. In: 2008 IEEE computer society conference on computer vision and pattern recognition workshops, IEEE, pp 1–8
84. Tang F, Wu X, Zhu Z, Wan Z, Yanchao C, Du Z, Gu L (2020) An end-to-end face recognition method with alignment learning. *Optik* 205:164238
85. Tarini M (2016) Volume-encoded UV-maps. *ACM Trans Graph (TOG)* 35(4):1–13
86. Tenenbaum JB, Silva V, Langford JC (2000) A global geometric framework for nonlinear dimensionality reduction. *Science* 290.5500:2319–2323
87. Tu X, Zhao J, Jiang Z, Yao L, Xie M, Zhao Y, He L, Ma Z, Feng J (2019) Joint 3d face reconstruction and dense face alignment from a single image with 2d-assisted self-supervised learning. *arXiv preprint arXiv:1903.09359* 1(2)
88. Turk M (2005) Eigenfaces and beyond. *Face Process Adv Model Methods* 55–86
89. Tzimiropoulos G, Pantic M (2014) Gauss-newton deformable part models for face alignment in-the-wild. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp 1851–1858
90. Vinay A, Hebbar D, Shekhar VS, Murthy KB, Natarajan S (2015) Two novel detector-descriptor based approaches for face recognition using sift and surf. *Procedia Comput Sci* 70:185–197
91. Vit P (2016) Comparison of various edge detection technique. *Int J Signal Process Image Process Pattern Recognit* 9:143–158
92. Wang YQ (2014) An analysis of the Viola-Jones face detection algorithm. *Image Process Line* 4:128–148
93. Wang Y, Li G, Ma L (2021) A sparse focus framework for visual fine-grained classification. *Multimed Tools Appl* 80:25271–25289
94. Wang H, Wang Y, Zhou Z et al (2018) Cosface: large margin cosine loss for deep face recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 5265–5274
95. Wenjing T, et al. (2017) Face recognition based on the fusion of wavelet packet sub-images and fisher linear discriminant. *Multimedia Tools Appl* 76.21:22725–22740
96. Wolf L, Hassner T, Maoz I (2011) Face recognition in unconstrained videos with matched background similarity. In: *CVPR, IEEE*, pp 529–534
97. Wu W, Qian C, Yang S, Wang Q, Cai Y, Zhou Q (2018) Look at boundary: a boundary-aware face alignment algorithm. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 2129–2138
98. Zhang W, Zhao X, Morvan J-M, Liming Chen (2018) Improving shadow suppression for illumination robust face recognition. *IEEE Trans Pattern Anal Mach Intell* 41(3):611–624
99. Zhao K, Xu J, Cheng M-M (2019) Regularface: deep face recognition via exclusive regularization. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp 1136–1144
100. Zhu X, Lei Z, Liu X, Shi H, Li SZ (2016) Face alignment across large poses: a 3d solution. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 146–155
101. Zhu X, Lei Z, Yan J, Yi D, Li SZ (2015) High-fidelity pose and expression normalization for face recognition in the wild. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 787–796
102. Zou G, Fu G, Gao M, Pan J, Liu Z (2020) A new approach for small sample face recognition with pose variation by fusing Gabor encoding features and deep features. *Multimed Tools Appl* 79(31):23571–23598

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.