mathematics

MDPI

Article

A Novel Face Frontalization Method by Seamlessly Integrating Landmark Detection and Decision Forest into Generative Adversarial Network (GAN)

Mahmood H. B. Alhlffee [1,*] and Yea-Shuan Huang [2]

[1] College of Computer Science and Electrical Engineering, Chung Hua University, Wufu Road, Hsinchu 300, Taiwan
[2] Department of Computer Science and Information Engineering, Chung Hua University, Wufu Road, Hsinchu 300, Taiwan; yeashuan@chu.edu.tw
* Correspondence: mahmood.bidir1985@gmail.com

Abstract: In real-world scenarios, posture variation and low-quality image resolution are two well-known factors that compromise the accuracy and reliability of face recognition system. These challenges can be overcome using various methods, including Generative Adversarial Networks (GANs). Despite this, concerns over the accuracy and reliability of GAN methods are increasing as the facial recognition market expands rapidly. The existing framework such as Two-Pathway GAN (TP-GAN) method has demonstrated that it is superior to numerous GAN methods that provide better face-texture details due to its unique deep neural network structure that allows it to perceive local details and global structure in a supervised manner. TP-GAN overcomes some of the obstacle associated with face frontalization tasks through the use of landmark detection and synthesis functions, but it remains challenging to achieve the desired performance across a wide range of datasets. To address the inherent limitations of TP-GAN, we propose a novel face frontalization method (NFF) combining landmark detection, decision forests, and data augmentation. NFF provides 2D landmark detection to integrate global structure with local details of the generator model so that more accurate facial feature representations and robust feature extractions can be achieved. NFF enhances the stability of the discriminator model over time by integrating decision forest capabilities into the TP-GAN discriminator core architecture that allows us to perform a wide range of facial pose tasks. Moreover, NFF uses data augmentation techniques to maximize training data by generating completely new synthetic data from existing data. Our evaluations are based on the Multi-PIE, FEI, and CAS-PEAL datasets. NFF results indicate that TP-GAN performance can be significantly enhanced by resolving the challenges described above, leading to high quality visualizations and rank-1 face identification.

Keywords: generative adversarial networks; landmark detection; decision tree and forest; data augmentation

MSC: 68T01

check for updates

Academic Editor: Konstantin Kozlov

Received: 27 December 2024
Revised: 25 January 2025
Accepted: 29 January 2025
Published: 2 February 2025

Citation: Alhlffee, M.H.B.; Huang, Y.-S. A Novel Face Frontalization Method by Seamlessly Integrating Landmark Detection and Decision Forest into Generative Adversarial Network (GAN). *Mathematics* **2025**, *13*, 499. https://doi.org/10.3390/math13030499

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Face recognition systems are highly sensitive to well-known factors, such as posture variation, low-quality image resolution, among other factors related to algorithm reliability. Detecting and identifying real faces becomes more challenging when posture variations are more pronounced. Typically, this occurs when a person is not facing the camera during

video or photo surveillance, which is referred to as an uncontrolled environment. Uncontrolled environments can lead to a wide range of postures and appearances, impacting the accuracy of frontal and near-frontal approaches. The above challenge can be overcome by developing a method that can infer more facial texture details (facial features). Advancement in texture details has improved our understanding of recovering the facial images and improved the generalization ability of face recognition through generative methods such as GANs. GAN uses multiple-task learning and feature representation techniques in order to identify an individual by a few facial features or using the entire faces without compromising their identity [1–4]. Various challenges prevent these methods from enhancing the frontalization samples and boosting recognition rates. Such challenges are caused by a variety of reasons, including those related to face rotations and datasets such that the method cannot adapt and learn to changes in posture variation under varying circumstances, the effect of low-quality image resolution using certain datasets, and insufficient data samples for a given individual in extreme poses [5–8]. Several methods have been developed to address these challenges. For instance, ref. [9] proposed Wasserstein GAN (WGAN) as a method for dealing with network training variability that leads to unwanted fine gradients in the discriminator function. Ref. [10] proposed Least Squares GAN (LSGAN) using least squares as a discriminator. Compared to GAN, LSGAN generates better quality images, stabilizes model performance, and accelerates convergence speed. LSGAN requires a large amount of face images for training in order to produce desired results. Ref. [11] proposed a TP-GAN model that combines adversarial loss, symmetry loss, and identity preservation loss to constrain this ill-posed problem. Instead of relying on intermediate features, the TP-GAN method leverages the synthesized identity-preserving image directly. Ref. [12] proposed a hierarchical GAN (HI-GAN) that can solve serious denoising problems by preserving high-frequency features like edges and textures through the use of a generator model. HI-GAN creates an effective adversarial loss function that improves discriminator stability. Ref. [13] presented a method for generating a fake face that maintains the basic structure and angle of a real face using facial landmarks. During pre- and post-processing, this method aims to maintain basic structural features of the real face by obtaining acceptable image quality from a desired face shape and angle. Ref. [14] dispensed a method for finding the latent common embedding subspace between profile face features and frontal face features based on a GAN embedded model. With this method, a given input image is transformed into its frontal face using a mapping function that adds residuals to the input image. Ref. [15] presented a novel method for learning pose-invariant feature representation without disentangling features or normalizing profile faces. This method allows the learning of pose-invariant feature embedding by distilling angular knowledge from teacher networks to student networks. Afterword, using Pose-Adaptive Angular Distillation loss mitigates the negative effects of unbalanced pose distribution in the training dataset. Ref. [16] proposed a framework for unsupervised face frontalization referred to as Disentangled Representation-Learning CycleGAN (DRCycleGAN). The DRCycleGAN can be trained by embedding face images in identity space and pose space, and then performing a paired forward and backward mapping (i.e., face frontalization). Such methods are well known and considered an adequate starting point because they can maintain some facial poses. In varying conditions, such as posture variation and low-quality image resolution, many GAN algorithms produce artifacts, such as sharpness, smoothness, blur, and clarity. Moreover, some existing GAN methods are quite complex because each model at each level is trained independently, affecting their performance because they do not receive updates from each other. Despite the impressive results of some GAN methods when it comes to generating face images, under large face poses, those methods produce images with lower diversity than images of the real one. Other approaches, like [17], penalize

outliers excessively, whereas those like [18] require significant time and resources and have limited capability in face generalization. The performance of alternatives such as [19] suffers from model collapse, weak neural structures, and insufficient balance training, while [20,21] produce fragments that closely resemble water droplets due to instance layer normalization, which makes it challenging to obtain accurate results. The limitations listed above are just a few of the many associated with existing face recognition algorithms that utilize GAN models. Face frontalization, such as TP-GAN [11], provides powerful methods for addressing some of the challenges involved in generating high-quality face images. However, the replication of TP-GAN across multiple datasets has three major limitations. First of all, it is important to point out that TP-GAN generates face images by transforming local texture details and global structure into feature maps, which can be influenced by posture variations, facial shapes (a few examples are provided in the method sections), and other factors that affect model reliability and visual adequacy. Second, the TP-GAN discriminator network is unable to distinguish between real and generated data due to its weak classifier architecture, which makes it difficult to decompose and interpret nonlinear data. In this case, both the generator and discriminator models will be unbalanced due to the inability to model nonlinear image distributions. Third, another determining factor that is found in datasets is the low-quality image resolution and insufficient sample quantity for a given individual in extreme poses, which leads to multi-factor failures, including the model's inability to adapt and learn to posture variations, difficulties in training behaviors associated with different datasets, and poor accuracy. Combining these factors can lead to undesirable results such as color bias between frontal synthesized faces and input profiles, further reducing recognition accuracy and generalization ability. Hence, the synthesized images cannot be used for specific facial analyses, such as recognition and verification. Therefore, our work is motivated by the fact that pose variations are highly susceptible to facial recognition systems.

Toward this end, we propose in this paper a novel face frontalization method (NFF) that addresses three major issues:

1. We introduce a 2D landmark detection to combine the global structure with local details so that more accurate facial feature representations and robust feature extractions can be obtained. The landmark detection method is introduced as a new technique for improving the TP-GAN generator model to incorporate empirical face pose and shapes into the learning process. A method like this enables the easy adaptation and learning of two-dimensional face models, enhancing texture details with target pose and shape variations.
2. We enhance the stability of the discriminator model by integrating decision forest capabilities into the TP-GAN discriminator core architecture network. Specifically, we are interested in the final layer of the discrimination network, usually referred to as the fully connected layer (FC). Consequently, this layer prevents the discriminator from easily interpreting or breaking down nonlinear data, hindering the ability to model joint distributions based on nonlinear data. Such a problem can be overcome by using the decision forest ability in a TP-GAN setup.
3. We utilize data augmentation techniques to maximize the training data by generating completely new synthetic data from existing data. Augmented data can provide an ideal solution to two popular problems, low-quality resolution and unbalanced face poses within datasets, which can affect curriculum learning accuracy and lead to high training errors.

2. The Proposed Novel Face Frontalization Method (NFF)

This section begins with a brief description of the TP-GAN architecture, followed by a detailed description of NFF.

2.1. TP-GAN Architecture

TP-GAN [11] consists of both generator (G) and discriminator (D) networks that work cooperatively end-to-end to synthesize frontal-face images as shown in Figure 1. The G network consists of local (l) and global (g) pathways that are based on a convolutional neural network (CNN) G_{θ_G} parameterized by θ_G. Each pathway consists of one down-sampling encoder G_{θ_E} and one up-sampling decoder G_{θ_D}. The local pathway G_{θ_l} is made up of four landmark patch networks $G_{\theta_i^l}, i \in \{0,1,2,3\}$, which represent the local textures around facial landmarks. On the other hand, the global pathway network ($G_{\theta_E^g}$,$G_{\theta_D^g}$) is used to provide global facial structures. The final synthesized image is constructed by merging the local textures details with the global facial structure, and it is used to perform the identity-preserving task using loss $L_{cross-entropy}$. The D network is based on a single CNN that distinguishes between synthetic frontal (SF) views and ground truth frontal (GT) views. More detailed description can be found in [11].

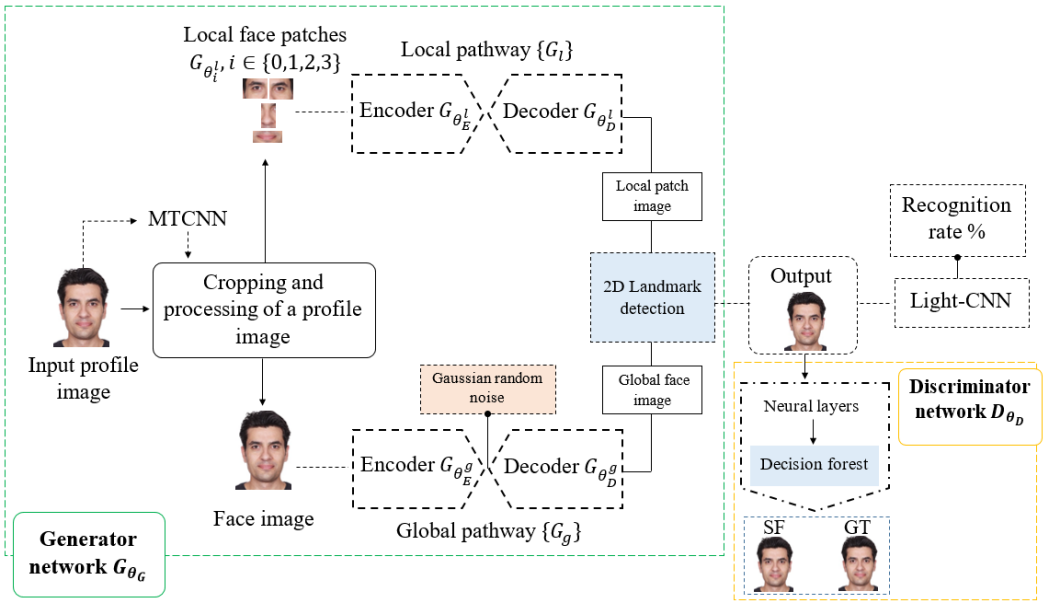


Figure 1. An overview of the NFF architecture. The G is based on two CNNs (local and global pathways), while the D employs a single CNN that distinguishes between synthetic frontal views and ground truth frontal views. Following that, the light CNN model is used to perform face identity recognition.

2.2. Generator with Landmark Detection

The first key contribution of this work lies in the integration of landmark detection into the TP-GAN global pathway structure, which enables better feature representation. This work utilizes the landmark detection mechanism [22–24], which has been proposed for 2D face landmark localization, to extract the most relevant features around the target face and to obtain an accurate frontalization image while maintaining as much of the facial structure as possible. The facial landmarks are the most compressed representations of a face that include facial structures, texture, and other details. Several key advantages of landmark detection have been noted in [25–28]. These methods have successfully aligned faces based on a structured relationship between landmark detection and the target face image. A method like this works well when certain datasets are selected without consideration

of other factors such as posture variation, facial shapes, and other related factors under various situations. Combining these factors can have a dramatic effect on the state-of-the-art face frontalization methods. The value of landmark detection lies in its ability to extract and integrate the most relevant features during the learning process. Figure 1 illustrates the generation of facial images. We begin by performing face detection using Multi-Task Cascade CNN [29]. A profile image is then cropped and processed. Four landmark patch networks capture the local texture details around four facial landmarks. An image of each patch is rotated through a set of filters to obtain the frontal view (after rotation, facial landmarks remain in the center). A multi-facial feature map is then used to combine the four facial tensors. The tensor features are placed at "template landmark location" and a max-out fusion strategy is used to minimize stitches on overlapping areas. Afterwards, zero padding is applied to the rows and columns surrounding each landmark template. However, the existing local landmark detection method used by TP-GAN cannot provide sufficient texture detail for faces that have different shapes or image with low resolutions, as each local synthesis face is generated with the same fixed patch (or template) centralized location, regardless of its shape or variance resolution. Typically, texture is defined by the variation in brightness intensity of pixels within an image. The texture level represents an attribute such as smoothness, coarseness, regularity, etc. The challenge arises when the local templates and global face structure are not cantered correctly, or when they differ in position/viewpoint, causing pixel distortions that can degrade texture detail capture. It is crucial to understand these challenges and adapt a method that eliminates them; otherwise, the results will be less accurate, so our method must be as robust and stable as possible. To address this problem, we design a second landmark localization method that utilizes the Face Alignment Network (FAN) based on the HourGlass (HG) architecture to model human poses to improve texture maps and reconstruct missing details. In landmark detection, the appearance spot of the mismatched or drifted face image is repositioned in order to recover the texture details of the synthesized face image. The term mismatched or drifted image refers to face images that do not correspond to the correct position or are inaccurately positioned (of an image pixel). Examples are illustrated in Figure 2. This study uses a landmark detection algorithm based on 68 points. A landmark serves as an indicator that repositions synthesized face patches of local pathway into the more accurate locations. Note: For each synthesized face patch, we perform further statistical analysis to gain insight into how many proportions of landmarks can be successfully detected for each patch component (i.e., left eye, right eye, nose, and mouth corners) by the second landmark detector. There are 68 landmark points in total, of which landmark indexes 37 to 41 are used to form a *left-eye* landmark patch, 43 to 48 are used to form a *right-eye* landmark patch, 28 to 36 are used to form a *nose* landmark patch, and 48 to 60 are used to form a *mouth* landmark patch. The above landmark points are derived from a facial image using Multi-Task Cascade CNN. We use Normalized Mean Error (NME) to measure the average distance between the profile face ID image landmark points (P^{FLP}), and the synthesized face image predicted landmark points (S^{PLP}). NME is calculated by computing the Euclidean Distance (ED) between the P^{FLP} and its S^{PLP} location:

$$NME = \sum_{n=1}^{4} \frac{||P_n^{FLP} - S_n^{PLP}||_2}{d_n} \quad (1)$$

Here, n represents landmark points, and d_n represents the normalizing distance (which means d_n:d_1 is the ED between the center of the right eye and the top of the nose, d_2 is the ED between the center of the left eye and the top of the nose, and d_3 and d_4 are the ED between top of the nose and the mouth corners). Every patch component has its average distance between landmark points normalized by a reference distance. Each patch

component has a reference distance, which is typically a fixed distance between predefined landmarks. The ED is used to measure the fixed distances between predefined landmark points of patch components. Afterward, we utilize a light CNN to compute the similarity of NME for each patch component. Table 1 shows the total average landmark detection rate error for each patch component. The process reveals how accurate our landmark detection is for each patch component, which is one of our primary concerns. Face patches serve as a unique feature that sets humans apart from other human face components based on geometric relationships. Many practical face recognition applications depend on such a process. Using landmarks is motivated by the fact that pose variations are highly susceptible to facial recognition systems. Changing any of the above facts can dramatically alter facial appearance, making face recognition less accurate. A landmark method provides us with valuable information about face regions, empowering our model to produce more realistic images.

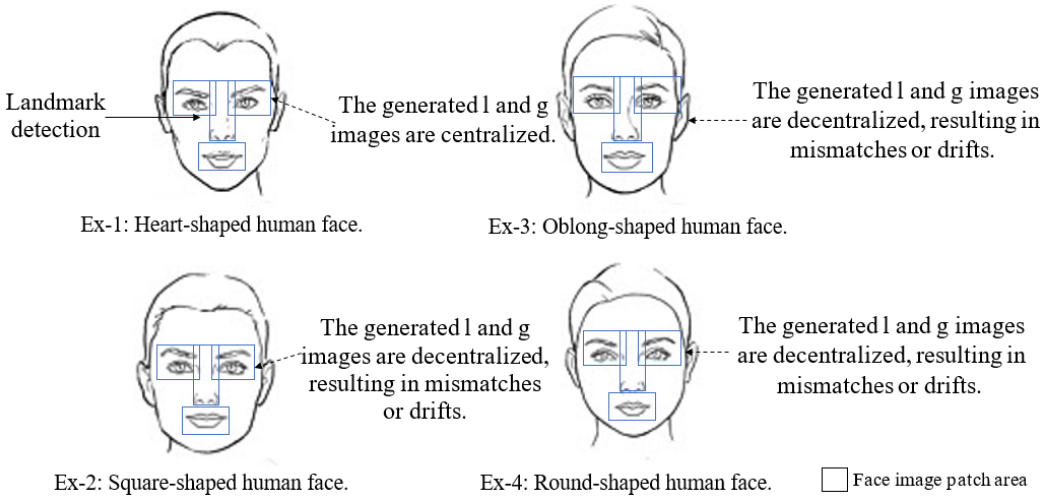


Figure 2. An overview of the different facial shapes. Example-1 looks very accurate because local (l) and global (g) synthesis images are centralized. The faces in example-2, -3, and -4 are inaccurate because the local and global synthesis images are generated decentralized, resulting in mismatches or drifts. The faces in example-2, -3, and -4 can be treated by our method regardless of their shape.

Table 1. A comparison of the NME of TP-GAN and NFF for each patch component.

Category List	Rul. H., et al. NME (%)	NFF NME (%)
Left-eye location	10.37%	8.92%
Right-eye location	14.35%	11.77%
Nose location	9.11%	7.10%
Mouth location	7.80%	6.70%

2.3. Discriminator with Decision Forest (DF)

DF has outperformed many existing classification methods due to its unique ability to discriminate between target values through binary rules. DF are composed of several decision trees. A DF prediction is essentially the sum of the predictions made by its trees. The use of DF in deep learning can take two forms: it can be incorporated into the core network architecture [30,31] or to influence the training process [32,33]. This work integrates DF into the core network architecture by replacing the final fully connected (FC) layer of the TP-GAN discriminator network. Our work is similar to [32], in which the soft function replaces the hard decision routing function with a different sigmoid function at every decision node. Unlike internal nodes, decision nodes redirect instances with a certain probability to their children. Each path leads to every leaf, and each leaf contributes to the

final decision but with varying probabilities. Our modified network architecture is shown in Figure 3.

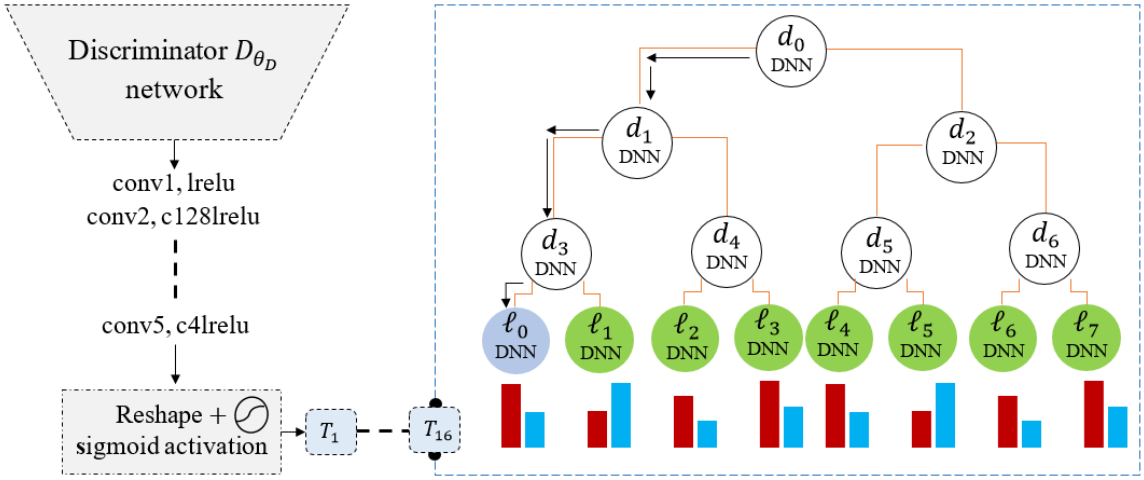


Figure 3. An overview of DF integration into the TP-GAN D-network, where d denotes the decision node and ℓ denotes the leaf node. A DF design consists of 16 trees of 9 depth levels each (i.e., $T_1, \cdots, T_{16}$). The arrows represent the paths used to route information of sample x along a tree to reach leaf ℓ_0, which has probability $\mu_{\ell_0} = d_0(x)\, d_1(x)\, d_3(x)$.

Through this work, we aim to produce representations of faces that retain texture details, face structure, and other variables, allowing for high-performance face classification accuracy in a variety of situations. The facts listed above are not disclosed or mentioned in [30–32,34,35]. It is quite impressive that these methods work well and achieve remarkable accuracy when dealing with datasets that do not pose any challenges. These methods are designed to handle frontal and near-frontal problems successfully using large amounts of training data. Compared to the above methods, our method may differ in some aspects and may not share the same characteristics. The DF is built by considering a variety of factors, such as unbalanced facial poses, network structure, low-resolution image quality, and other related classification problems.

2.4. Data Augmentation (DA)

Datasets used to train deep learning frameworks usually contain a large number of frontal or near-frontal images. The absence of large face postures contributes to poor model performance when training these deep learning frameworks. A promising solution to these problems is GAN, which creates new synthetic data from existing data [2,4,36,37]. Augmenting the dataset can serve as an effective resource to improve TP-GAN in both task generalization and classification. Training with large posture face samples can minimize over-fitting among layers as TP-GAN learns and adapts to more details over time. Compared to previous methods, this work has several unique features (such as the large number of training face pose samples, the use of light CNN for face identification, and the use of human observation for label verification). We aim to improve the generalization abilities of the generator model by training it several times with varying pose samples (±15° to ±90°). The generated face samples are then used to build and train a DF model. We aim to strengthen the discriminator network and build an accurate classifier model to enable better accurate face identification and distinguishing real faces from generated ones. Despite their impressive performance, DF still has some limitations when used alone to solve specific classification problems. DF is capable of handling high-quality faces with adequate samples but struggles with unbalanced-class poses with insufficient face samples. In this case, DF would have limited classification abilities without many face samples to

learn from. Large face image samples data can be used to train a classifier model to be highly accurate by selecting appropriate features and decision-making capabilities. Our DA strategy eliminates the DF limitations. Hence, Table 2 shows the amount of data for each face pose, followed by a description of each step.

1. Step-1: The purpose of this step is to train the DF with Multi-PIE, FEI, and CASE-PEAL datasets. We aim to train the DF using different face pose samples 15° to 90°, in order to provide a well-balanced data structure that can adapt and learn from changes in facial features.
2. Step-2: Utilize the TP-GAN for DA to generate synthetic data to augment a face pose dataset (Set 1). The goal of this step is to train TP-GAN on a variety of balanced-pose synthetic images, in order to enable it to learn and adapt more to facial features. In other words, large-pose faces generate more face images than small-pose faces.
3. Step-3: In this step, the DF model is trained to learn more facial features by combining the original face samples (Set 2) with the face pose samples generated from step-2.
4. Step-4: Replace the final fully connected layer of the light-CNN discriminator with DF and train the model simultaneously. In this step, the same training samples (Set 2) are used as in step-3. TP-GAN will generate new samples (Set 3) for each face pose.
5. Step-5: The DF will be trained on all samples from Set 2 and Set 3. After training, the TP-GAN generator will generate a new set of samples (Set 4). We would like to point out that both the samples generated in step-4 and -5 are classified by a light CNN prior to collection. Samples are collected only if they are correctly classified; otherwise, they are discarded. Such a process is to ensure clean augmented data and maintain identity.
6. Step-6: All samples from Sets 2, 3 and 4 will be used for DF training.

Table 2. The quantity of face images that are synthesized by the TP-GAN model for each dataset.

Dataset Types	Posture Range	The Total Number of Available Face Poses Before Data Augmentation	The Total Number of Available Face Poses After Data Augmentation
Multi-PIE	15° to 90°	2908	7098
FEI	15° to 90°	2800	6753
CAS-PEAL	15° to 90°	12,000	35,988

Finally, we would like to mention that DA was conducted using TP-GAN and MTCNN to provide better classification features. MTCNN has been widely adopted for facial feature alignment and detection, as well as for a variety of other applications. Despite potential inconsistencies or over-smoothing caused by factors beyond our control, DA remains effective in incorporating pose information.

2.5. Adversarial Networks

GAN [38] consists of generator G_{θ_G} and discriminator D_{θ_D} models. Let us begin with a brief discussion of some technical aspects of generating a frontal face. The goal of frontal view synthesis is to generate a photorealistic and identity-preserving frontal view image (I^F) from a given image under pose variation, i.e., a profile image (I^P). In order to train such networks, a corresponding pair (I^F and I^P) is required from multiple identities y. I^F and I^P both operate on a pixel space that has a dimensional size of $W \times H \times C$. This section omits the encoder and decoder structures since they have already been explained in Section 2.1. The optimization of network parameters G_{θ_G} begins with the minimization of

the synthesis loss (L_{syn}) and the aforementioned $L_{cross-entropy}$. The optimization problem can be expressed as follows using N pairs of $\{I_n^F, I_n^P\}$ training pairs:

$$\hat{\theta}_G = \frac{1}{N} \arg\min_{\theta_G} \sum_{n=1}^{N} \{L_{syn}(G_{\theta_G}(I_n^P), I_n^F) + \alpha L_{cross-entropy}(G_{\theta_E^g}(I_n^P), y_n)\}. \quad (2)$$

Here, α refers to the weighting parameter, whereas L_{syn} represents the weighted sum of all individual losses that constrain the image to exist within the desired manifolds. Section 2.5.1 will provide detailed descriptions of each loss function.

Well-balanced and effective G and D models are necessary to generate high-quality face images. Essentially, if G is not good enough, we will not be able to fool D, which leads to unreliable convergence. Poor D would also classify nonsensical images as real, which would prevent the model from accurate training, and we would never obtain the desired results. A game-theory approach leads to the following objective function:

$$\min_{\theta_G} \max_{\theta_D} = [\mathbb{E}_{I^F} \backsim_{P(I^F)} logD_{\theta_D}(I^F) + \mathbb{E}_{I^P} \backsim_{P(I^P)} log(1 - D_{\theta_D}(G_{\theta_G}(I^P)))]. \quad (3)$$

We can perform gradient descent on the objective function by maximizing D, while minimizing G allows us to compute an objective function using gradient descent. It is therefore necessary to alternate gradient ascent and descent when training G and D:

$$\max_{\theta_D} [\mathbb{E}_{I^F} \backsim_{P(I^F)} logD_{\theta_D}(I^F) + \mathbb{E}_{I^P} \backsim_{P(I^P)} log(1 - D_{\theta_D}(G_{\theta_G}(I^P)))]. \quad (4)$$

Gradient ascent on D

$$\min_{\theta_G} \mathbb{E}_{I^P} \backsim_{P(I^P)} log(1 - D_{\theta_D}(G_{\theta_G}(I^P))). \quad (5)$$

Gradient descent on G

D can maximize adversarial networks, allowing us to ascend gradients; whereas G can minimize adversarial networks, allowing us to descend gradients. The gradients provided by Equation (3) might not be sufficient for G to learn well. Therefore, D can reject samples with high confidence in the early stages of learning when G is poor since they are clearly different from the training data. In this case, $log(1 - D_{\theta_D}(G_{\theta_G}(I^P)))$ saturates. A better alternative to training G to minimize $log(1 - D_{\theta_D}(G_{\theta_G}(I^P)))$ is to train G to maximize $logD_{\theta_D}(I^F)$. As a result of using such an objective function, G and D will have much stronger gradients at the beginning of the learning process.

It is common for a single D network to overemphasize certain features in order to fool the current G network, leading to drift and distorted results. Thus, local patches sampled from the refined image should have similar statistics to real image patches. An alternative to defining a global D network is to define one that classifies each local image patch separately. Using such divisions will result in more samples per image, which will improve learning as well as eliminating drift and artifacts.

2.5.1. Comprehensive Loss Functions

Using a combined loss function allows us to take advantage of both the frontal face distribution and pre-trained discriminative models to infer the frontal views from profiles while preserving the identity of the individual. The synthesized identity-preserving image is directly used for downstream tasks such as face recognition and attribution estimation, unlike previous deep learning methods that rely primarily on intermediate features.

To begin, we will define each symbol, followed by a description of each loss function. Let I represent an output image, W and H represent the width and height of I, respectively,

(x, y) represent the 2D image pixel coordinate, I^{pred} represent a predicted (i.e., synthesized) frontal-face image of I, and I^{gt} represent a frontal-face image of the ground-truth category of I. I^{gt}_{local} is the composition image of the local facial patches of I^{gt}, and I^{pred}_{local} is the composition image of the local facial patches of I.

- Pixel-wise loss (L_{px}):

A L_{px} is used at multiple locations to ensure the consistency of image content across scales:

$$L_{px} = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} |I^{gt}(x, y) - I^{pred}(x, y)|. \tag{6}$$

Even though L_{px} may cause artifacts that adversely affect the results, they are a prerequisite for accelerated optimization and superior performance.

- Symmetry loss (L_{sym}):

A L_{sym} constraint imposed on the synthesized images may effectively alleviate the self-occlusion problem and thus greatly improve performance for large pose cases. The L_{sym} is robust to illumination changes both in pixel space and Laplacian image space:

$$L_{sym} = \frac{1}{W/2 \times H} \sum_{x=1}^{W/2} \sum_{y=1}^{H} |I^{pred}(x, y) - I^{pred}(W - (x - 1), y)|. \tag{7}$$

The symmetry loss function is used to calculate the symmetry of the synthesized face image.

- Identity preserving loss (L_{ip}):

Developing the "recognition via generation" framework requires preserving identity while synthesizing the frontal view image. A loss function of this kind ensures that the prediction is close to the ground truth in the compact deep feature space:

$$L_{ip} = \sum_{l=1}^{2} \frac{1}{W_l \times H_l} \sum_{x=1}^{W_l} \sum_{y=1}^{H_l} |F_l^{gt}(x, y) - F_l^{pred}(x, y)|. \tag{8}$$

Here, F_l^{gt} and F_l^{pred} corresponds to the feature maps of the last two layers I^{gt} and I^{pred}, while W_l and H_l correspond to the width and height of the feature map of the last $l - th$ layer ($l = 1, 2$) layer. These layers are all components of light CNN [11]. The L_{ip} is used to ensure that a prediction in a deep feature space is relatively close to ground truth.

- Adversarial loss (L_{adv}):

An adversarial loss distinguishes between synthetic and real frontal images, and can be calculated as follows:

$$L_{adv} = \frac{1}{N} \sum_{n=1}^{N} -logD_{\theta_D}(G_{\theta_G}(I_n^P)). \tag{9}$$

This kind of loss function is useful for achieving visually satisfying or appealing results across real frontal-face images I^F and synthesized frontal-face images $G_{\theta_G}(I_n^P)$.

- Total variation loss (L_{tv}):

The L_{tv} reduces noise and pixels in generated images, or generates images with the style of one and the content of another:

$$L_{tv} = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} |I_{x,y}^{pred} - I_{x-1,y}^{pred}| + |I_{x,y}^{pred} - I_{x,y-1}^{pred}|. \quad (10)$$

The use of L_{tv} can alleviate the problem of unfavorable visual artifacts that are caused by deep neural network models which adversely affect visualization quality and recognition accuracy.

- Classification loss ($L_{classify}$):

The classification loss is designed to minimize errors and improve overall model performance:

$$L_{classify} = -\sum_{i} y_i^{true} log_2(y_i^{perd}). \quad (11)$$

where i refers to the class index, y^{true} refers to the tensor of the one-hot true target of i, and y^{perd} refers to the predicted probability tensor. $L_{classify}$ ensures that the synthesized frontal-face image can be classified correctly using cross entropy loss.

- Local pixel-wise loss (L_{local}):

L_{local} is used to recover image structures as well as estimating the facial image background:

$$L_{local} = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} |I_{local}^{gt}(x,y) - I_{local}^{pred}(x,y)|. \quad (12)$$

The purpose of this loss is to calculate the total average pixel difference between I_{local}^{gt} and I_{local}^{pred}.

- Generator loss functions ($L_{generator}$):

Based on the weighted sum of the above loss functions, the $L_{generator}$ is defined as follows:

$$L_{generator} = L_{px} + \lambda_1 L_{sym} + \lambda_2 L_{ip} + \lambda_3 L_{adv} + \lambda_4 L_{tv} + \lambda_5 L_{classify} + \lambda_6 L_{local}. \quad (13)$$

where $\lambda_i (i = 1, 2, \ldots, 6)$ represent weights that coordinate the different losses, which are set to ($\lambda_1 = 0.1, \lambda_2 = 0.001, \lambda_3 = 0.005, \lambda_4 = 0.0011, \lambda_5 = 0.1$, and $\lambda_6 = 0.3$) throughout our experiments. $L_{generator}$ are used to guide the identity-preserving inference of the frontal view synthesis. On the other hand, the L_{adv} pushes the generative model forward to produce a satisfactory frontalization image.

2.5.2. Decision Trees and Forest

Let $x \in X$ denote the features variable set $(x_1, \ldots, x_n)$, where X belongs to the total feature set of x, and $y \in Y$ denotes the target variables set $(y_1, \ldots, y_n)$, where Y belongs to the total variable set of y. Decision trees consist of two types of nodes: decision nodes, denoted by N_d, and leaf nodes, denoted by N_ℓ. Here, N denotes a tree node and d denotes a decision node, whereas ℓ denotes a leaf node, while the root node represents the starting point within the tree from which we divide the data into two or more sets. N_d is responsible for routing data down the tree by sending them to either the $left$ or $right$ child nodes, while ℓ holds the prediction distribution. Nodes N_d route data downward in the tree, and may be referred to as decision nodes, split nodes, or internal nodes. N_ℓ, on the other hand, hold prediction distributions, which do not further classify the examples but simply categorize them within each node. It is worth noting that each $\ell \in N_\ell$ holds a probability distribution π_ℓ over Y. Each $n \in N_d$ is paired with a decision function $d_n \colon (x; \theta) \to \{0, 1\}$ parametrized by θ, which determines that the samples will be routed to the $left$ child node when d_n is 0 and to the $right$ child node when d_n is 1. An input sample x is mapped from the root

node via N_d to the leaf node: $\ell = N_d(x;\theta)$. Whenever the sample reaches ℓ, the class-label distribution π_ℓ provides a prediction of the related tree.

- Decision trees (DT):

DT combines the leaves' values based on their proportions to arrive at a single prediction. In this case, N_d returns a value representing the proportion of the left and right sub-trees:

$$d_n(x;\theta) = \sigma(f_n(x;\theta)). \tag{14}$$

where $\sigma(x)$=$(1+e^{-x})^{-1}$ contributes as a sigmoid function, while $f_n(x;\theta) : X \to \mathbb{R}$ contributes as a true value function based on x and θ, while f_n function contributes as a linear output unit with deep neural network (DNN) capabilities, which then uses the σ function to achieve its output within a range of $[0,1]$. Tree nodes are designed with DNN structures, allowing the classifier to learn the capabilities of different classes with a minimum effort. In order to determine each leaf proportional contribution to the final tree output, we define $\mu_\ell\ (x;\theta)$ as follows:

$$\mu_\ell(x|\theta) = \prod_{n \in N_d} d_n(x;\theta)^{1_{\ell \swarrow n}} \bar{d}_n(x;\theta)^{1_{n \searrow \ell}}. \tag{15}$$

where $\bar{d}_n(x;\theta) = 1 - d_n(x;\theta)$. 1_C acts as an indicator function and is set to 1 when C is met, and 0 otherwise, $\ell \swarrow n$ refers to left n node, while $n \searrow \ell$ refers to right n node.

All $\mu_\ell(x|\theta)$ but one has the value of 1, while the rest have the value of 0. Despite the fact that the product in Equation (15) applies to all nodes, only the decision leading to ℓ contributes to μ_ℓ. A good example can be found in Figure 3. In so doing, we arrive at the following prediction value:

$$Q(y|x,\theta,\pi) = \sum_{\ell \in N_\ell} \pi_{\ell_y} \mu_\ell(x|\theta). \tag{16}$$

All the parameter values for all N_d and N_ℓ are represented by θ; meanwhile, π_{ℓ_y} denotes the probability of a sample reaching ℓ to take on class y, while $\mu_\ell(x|\theta)$ presents as a routing function providing the probability of ensuring that sample x will reach ℓ. DT leads to superior model performance, allowing gradients to be generated easily by error back-propagation, which results in easy model updates.

- Decision forest (DF):

Unlike [39], our N_d are not strictly split into $left$ and $right$ nodes since differential binning can split a N_d into a multiple ℓ. The most relevant input features are taken into account in order to make complex data distributions easier to understand. The DF consists of a set of DT that performs as a forest $DF = (T_1, T_2, \cdots, T_N)$ that uses a majority voting mechanism, which can be calculated as follows:

$$F[y|x] = \sum_{n=1}^{N} Q(y|x,\theta_n,\pi_n). \tag{17}$$

The number of trees is determined by N, and the prediction value is determined by Q for each tree. With our approach, both forward and backward training can be carried out. In this fashion, we are able to train forest supervised, with N_ℓ and N_d both being updated concurrently.

- Learning tree nodes:

The back-propagation method for trees involves estimating the leaf predictions π and decision nodes θ. For the given dataset $\mathcal{T} \subset X \times Y$, the principle of minimum empirical risk (R) is applied, i.e., calculating the minimizer, which can be expressed as follows:

$$R(\theta, \pi; \mathcal{T}) = \frac{1}{|\mathcal{T}|} \sum_{(x,y)\in\mathcal{T}} L(\theta, \pi; x, y). \tag{18}$$

L is a weighted cross-entropy loss (weighted by its target distribution π and the path probability $\mu_\ell(x|\theta)$) corresponding to dataset samples $(x, y) \in \mathcal{T}$, which further can be expressed as follows:

$$L(\theta, \pi; x, y) = -\log(Q(y|x, \theta, \pi)). \tag{19}$$

Please refer to the (decision tree) section for symbol details. A method for minimization Equation (18) will be described in the next paragraph.

Equation (14) parameterizes each decision function by the same θ, which further parameterizes f_n. In the absence of assumptions about the type of functions in f_n, optimizing risk in relation to θ for a given π may not be a challenging task. To minimize the risks associated with θ, we use Stochastic Gradient Descent (SGD), which is commonly used in DNN:

$$\theta^{(t+1)} = \theta^{(t)} - \frac{\eta}{|\beta|} \sum_{(x,y)\in\beta} \frac{\partial L}{\partial \theta}(\theta^{(t)}, \pi; x, y). \tag{20}$$

where t indicates the time step, $0 < \eta$ indicates the learning rate, β indicates the mini-batch, and $\beta \subseteq \mathcal{T}$ indicates a subset of the training set samples. Based on the chain rule, we can express a gradient loss (L) with respect to θ as follows:

$$\frac{\partial L}{\partial \theta}(\theta, \pi; x, y) = \sum_{n\in N_d} \frac{\partial L(\theta, \pi; x, y)}{\partial f_n(x; \theta)} \frac{\partial f_n(x; \theta)}{\partial \theta}. \tag{21}$$

A DT can be used to calculate the L term as follows:

$$\frac{\partial L(\theta, \pi; x, y)}{\partial f_n(x; \theta)} = d_n(x; \theta) A_{n_r} - \bar{d}_n(x; \theta) A_{n_l}. \tag{22}$$

Here, n_r represents a *right* node child and n_l represents a $left$ node child. As for A_m, it is defined for generic nodes $m \in N$ that can be expressed as follows:

$$A_m = \frac{\sum_{\ell\in N_m} \pi_{\ell_y} \mu_\ell(x|\theta)}{Q(y|x, \theta, \pi)}. \tag{23}$$

In this case, N_m denotes the ℓ number at node m in the sub-tree. A detailed explanation of Equation (23), which describes an efficient method for computing A_m for all nodes m, is provided (Implementation details). We now address the problem of minimizing Equation (18) in relation to π under the same decision function θ as in the previous paragraph:

$$\min_{\pi} R(\theta, \pi; \mathcal{T}). \tag{24}$$

It is likely that the above equation will result in convex optimization problems that can be solved by identifying a global solution that can be easily recovered. Such convex optimization problems occur only at the level of a single node in [40]. It is important to emphasize that we consider the entire T where all predictions are summed up across all ℓ. A global minimizer of Equation (24) is computed by updating the distribution of N_ℓ nodes in iteration $t + 1$ for all $\ell \in N_\ell$ and $y \in Y$:

$$\pi_{\ell_y}^{(t+1)} = \frac{1}{P_\ell^{(t)}} \sum_{(x,y)\in\beta} \frac{\pi_{\ell_y}^{(t)} \mu_\ell(x|\theta)}{Q(y|x, \theta, \pi^{(t)})}. \tag{25}$$

where $P_{\ell}^{(t)}$ is a normalizing factor ensuring that $\sum_y \pi_{\ell_y}^{(t+1)} = 1$. The starting point can be any value as long as all elements are positive. The uniform initial distribution would be used in most cases, i.e., $\pi_{\ell_y}^{0} = |Y|^{-1}$.

- Training DF:

So far, the discussion has been limited to a single DT. The next step is to discuss an ensemble of F where every tree T shares the same structure. Based on the current estimates of θ, updating the prediction nodes independently becomes an easy process because each T in F has its own ℓ parameter π.

Instead of selecting a T in F for every β, we use an update mechanism similar to learning tree nodes. This type of update strategy is similar to dropout regularization [41], which updates specific network typologies based on SGD. A mechanism such as this can reduce the computation workload by updating individual T.

- Implementation details:

Decision nodes: Real-valued functions defined by $f_n(x;\theta)$ are not necessarily independent. The feature learning capabilities of T_s are implemented by embedding functions f_n into a CNN model with θ. Specifically, f_n is a DNN linear output unit which produces a probability routing decision after σ activation is applied to d_n in order to generate an output of $[0, 1]$. An output unit in a DNN does not make predictions directly; instead, it makes its own decisions to drive the d_n in a F. The final outcome of a DNN depends on the mixture of predictions ℓ derived from T routing decisions as x passes through it. Based on linear and independent assumptions, we achieve recovery similar to oblique [42] applied for the forest model through separate parameterization θ functions $f_n(x;\theta_n) = \theta_n^{\top} x$.

Routing function: In order to perform the routing function μ_{ℓ}, we are required to traverse T once. Let n_r and n_l indicate the right and left children of n node, respectively. Using the breadth-first approach, we set $\mu_{n_l} = d_n(x;\theta)\mu_n$ and $\mu_{n_r} = \bar{d}_n(x;\theta)\mu_n$. The values of the routing function that travel through T can be read from the ℓ at the end of T.

Learning decision nodes: For each sample (x, y) in β, we calculate the routing function $\mu_{\ell}(x;\theta)$ and the T prediction $(Q(y|x,\theta,\pi))$. Equivalent to Equation (22), each sample in β requires a backward gradient calculation. T can be traversed once from bottom to top, with the following starting point:

$$A_{\ell} = \frac{\pi_{\ell_y}\mu_{\ell}(x|\theta)}{Q(y|x,\theta,\pi)}. \tag{26}$$

All $\ell \in N_{\ell}$ have their T visited in reverse breadth-first order (bottom-to-top manner). The partial derivative of Equation (22) allows us to calculate A_{n_r} and A_{n_l}, assuming that the parent can read the children's A_{n_r} and A_{n_l}. This will be indispensable to the parent for the calculation of $A_n = A_{n_r} + A_{n_l}$.

3. Experiments

Our evaluation involves two highly regarded factors: the face frontalization quality results and the face identity preservation accuracy.

3.1. Experimental Settings

A single dataset may not provide reliable insight into our method, but different datasets can provide a reliable source that can be used to compare it with TP-GAN, as well as other state-of-the-art approaches. TP-GAN and NFF are therefore trained and evaluated on the Multi-PIE, FEI, and CAS-PEAL datasets. The Multi-PIE [43] dataset consists of 75,000 images of 337 individual faces that were captured under controlled circumstances

in a variety of postures, illuminations, and expressions. Each individual face has 15 poses and 20 illumination levels ranging from −90 to +90. The average age of the subjects is 27 years old, and they come from different ethnic backgrounds, with 60% European-Americans, 35% Asians, and 5% African-Americans. FEI [44] is a Brazilian facial dataset that includes 200 images per individual, resulting in 2800 images in total. The profile of every individual face image is rotated right or left 90 degrees in an upright frontal position. Each face has its own distinct feature, hairstyle, adornment, etc. CAS-PEAL [45] contains 99,594 gray-scale images of 1040 individuals (595 males and 445 females) with Asian faces. The dataset includes 27 images of faces in three different postures, such as straight ahead, up, down, and sideways. Moreover, the database contains 5 facial expressions, 6 accessories, and 15 illumination variations.

3.2. Face Frontalization Quality Results

Most face frontalization methods miss high-frequency facial details due to extreme poses. We demonstrate that NFF generates more realistic frontalized faces with rich texture detail and facial structure, a substantial improvement over TP-GAN, outperforming the state-of-the-art methods. The results are divided into two parts; the first shows TP-GAN and NFF synthesized face image results in Figures 4–6 and the second shows NFF synthesized face image in Figures 7–10 in comparison with state-of-the-art methods.

A comparison of Figures 4–6 advocates that the TP-GAN results are still accompanied by noise and lack effective supervision to decode texture information, leading to inferior generalization abilities as compared to NFF when it comes to visual quality and diversity. NFF allows us to generalize realistic-looking faces under various poses conditions. TP-GAN demonstrates limited capabilities for handling many posture regions and somewhat disappointing results for multiple-yew poses in Figure 6. NFF is able to restore high-quality areas, such as facial structure with fine details, which can capture subtle variations in facial posture better than TP-GAN. Furthermore, NFF exhibits a higher level of texture detail in facial images than TP-GAN, resulting in images that are more stable and recognizable. NFF appears to have better resolution and higher visual quality for many pose variations than TP-GAN, which performs poorly in generating high-quality face samples.

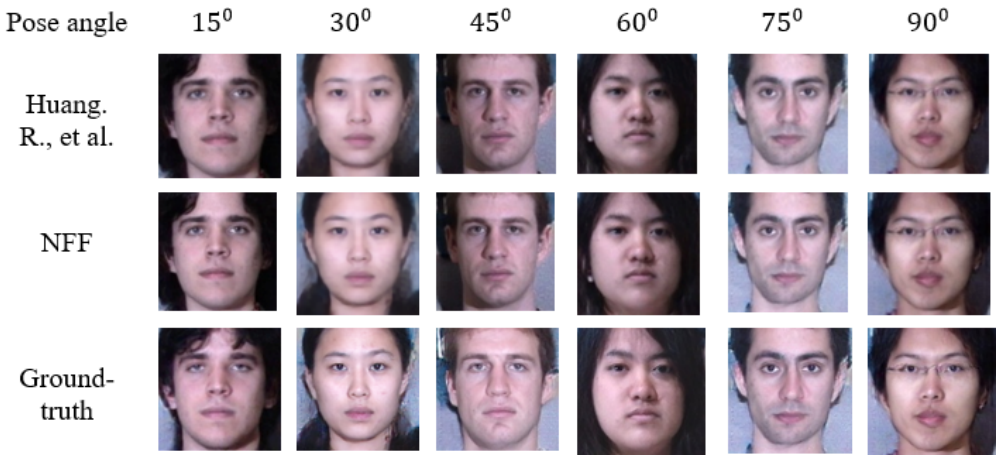


Figure 4. A comparison of the facial images produced by NFF with those produced by TP-GAN using the Multi-PIE dataset. Data were downloaded from the TP-GAN GitHub repository: https://github.com/HRLTY/TP-GAN, accessed on 4 August 2017.

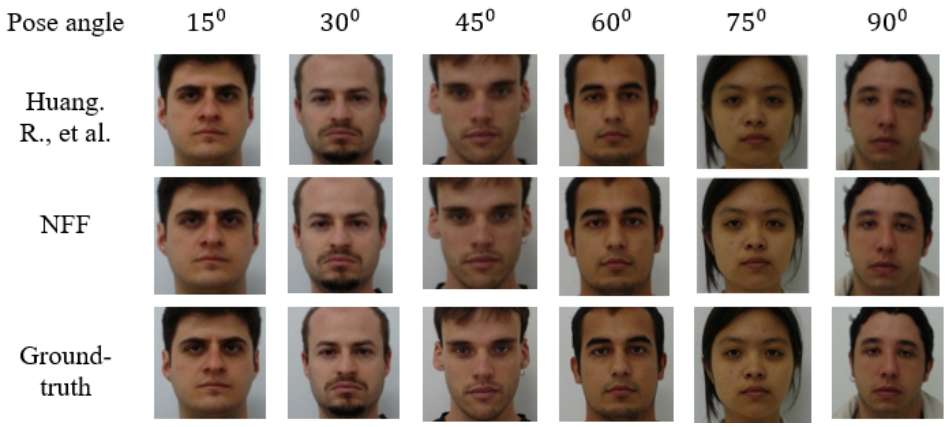


Figure 5. A comparison of the facial images produced by NFF and TP-GAN using the FEI dataset. The dataset was downloaded from the official repository of the FEI: https://fei.edu.br/~cet/facedatabase.html, accessed on 1 June 2005 and 1 March 2006.

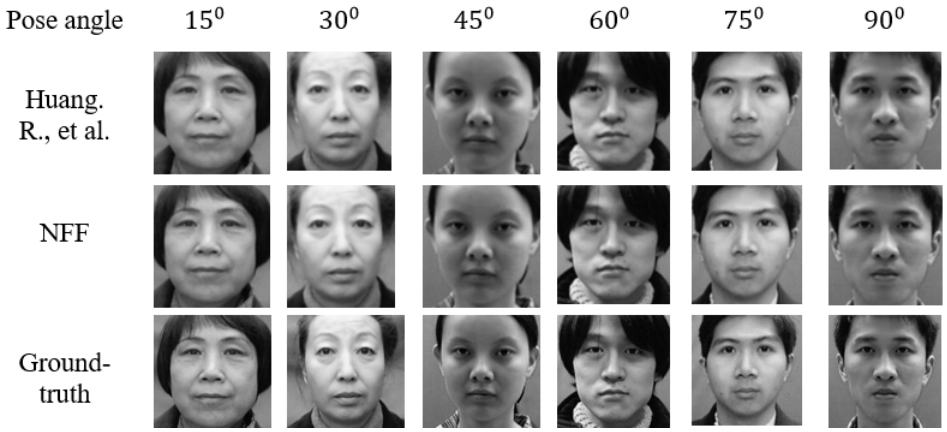


Figure 6. A comparison of the facial images produced by NFF with those produced by TP-GAN using CAS-PEAL dataset. The dataset was downloaded from the official CAS-PEAL repository at: https://github.com/YuYin1/DA-GAN, accessed on 1 February 2008.

For the second part, Figures 7–10 show a comparison of NFF and several state-of-the-art face frontalization methods. Despite the diversity of pose variations conditions, NFF is still capable of producing photorealistic images. Figure 7 presents a comparison of synthetic images generated by various methods using Multi-PIE. NFF is robust to extreme facial poses, making it a good candidate for face synthesis with posture variation. The synthesized images of [17,46] differ significantly from the originals. The lack of details makes it impossible to restore the most relevant texture details and produce realistic images. In [11,19], face images are recovered with severe blurring and failures. In [45], we show synthesized face images resulting from slight pose challenges, which may appear less photorealistic. Due to this failure, the face image lacks realistic objects and exhibits artifacts. We may not conclude that these methods are able to generate high-quality face images under every posture situation since they have shown low capabilities in this regard. However, NFF is able to recover most of the face, where other methods have shown disappointing outcomes and poorly structured facial details. Despite differences in the image color in Figure 8 [6,7,11,20,47,48] and Figure 10 [49], we consistently achieved higher texture detail across all faces. Most methods failed to produce satisfactory results, so their effectiveness cannot be assured until further investigation is undertaken to determine why such results were obtained and how they can be improved.

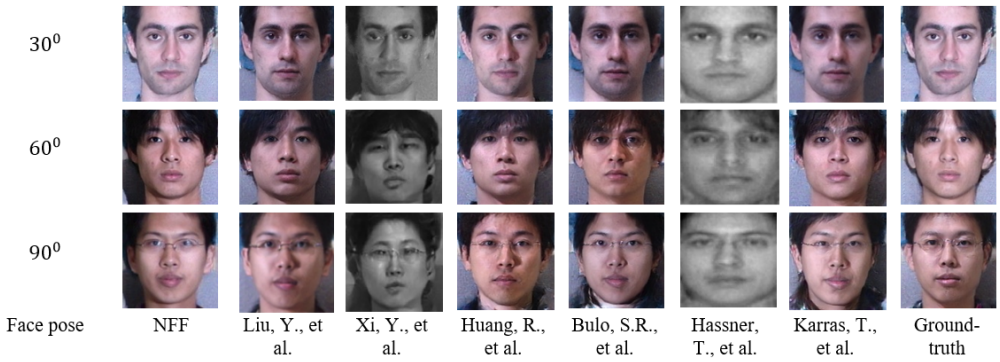


Figure 7. The frontal-profile synthesis results of NFF are compared to those of other methods using the Multi-PIE dataset. We downloaded the dataset from the TP-GAN GitHub repository: https://github.com/HRLTY/TP-GAN, accessed on 4 August 2017.

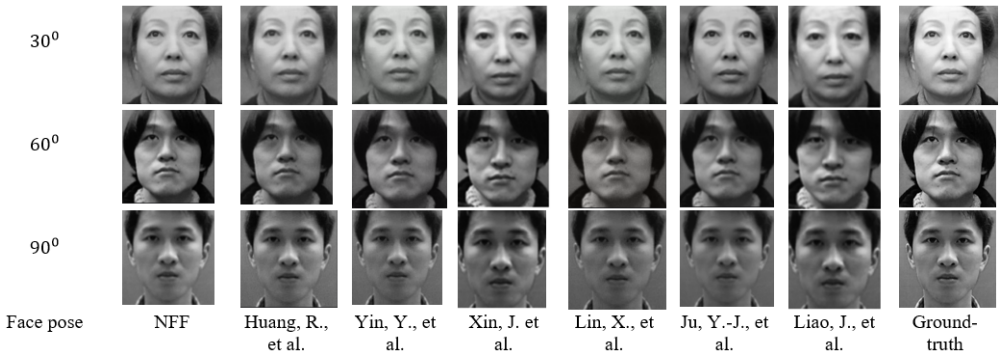


Figure 8. NFF frontalization faces are compared with those of other methods on the CAS-PEAL dataset. The dataset was downloaded from the official CAS-PEAL repository at: https://github.com/YuYin1/DA-GAN, accessed on 1 February 2008.

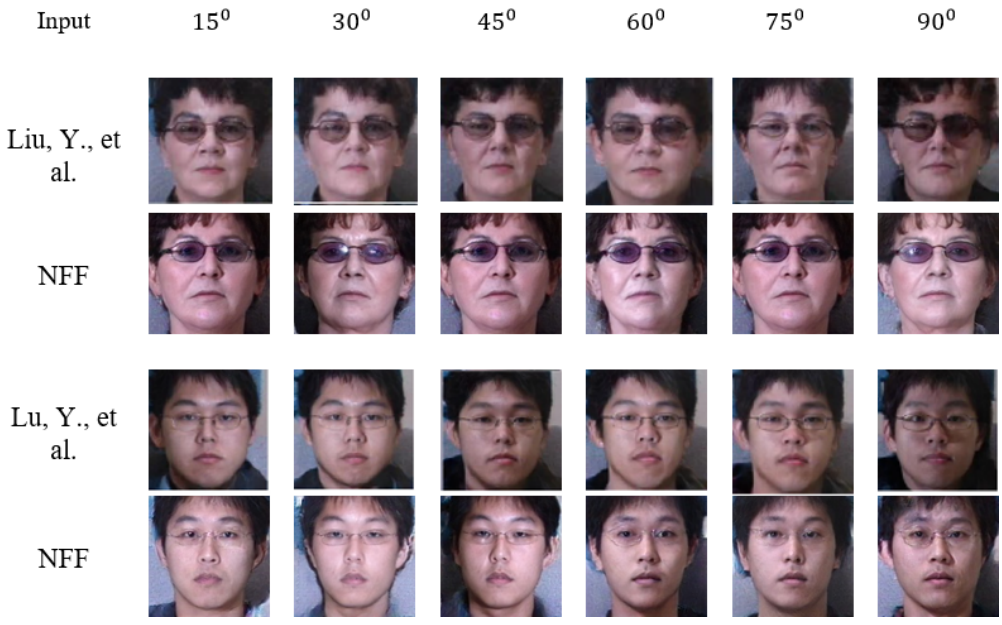


Figure 9. The frontalization faces of NFF are compared to other method using the Multi-PIE dataset. We downloaded the dataset from the TP-GAN GitHub repository: https://github.com/HRLTY/TP-GAN, accessed on 4 August 2017.

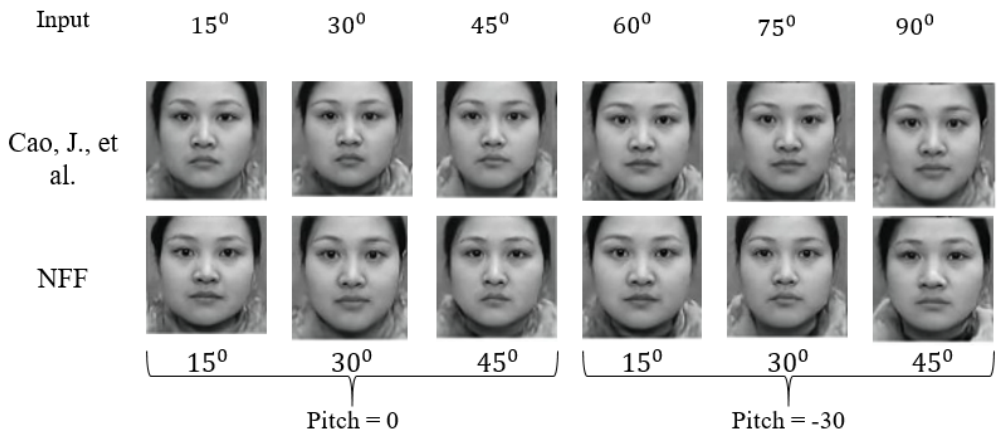


Figure 10. The frontalization faces of NFF are compared to other methods on the CAS-PEAL dataset. The dataset was downloaded from the official CAS-PEAL repository at: https://github.com/YuYin1/DA-GAN, accessed on 1 February 2008.

Since most methods failed to produce satisfactory results, their effectiveness cannot be assured until further investigation is undertaken to determine why such results were obtained and how they can be improved. In general, rendering a human face with a posture over 60° is considered challenging. For most methods, posture variation and low-quality resolution among many other issues are particularly challenging since significant semantic information is lost. Therefore, we observe some failures, as illustrated in Figure 9, where an individual wearing sunglasses fails to synthesize high-quality faces using the methods of [50,51]. We can see that the faces in [50,51] are very blurry and distorted when compared to NFF generated faces, which seem more realistic. This demonstrates that outstanding results can be achieved with enough training data and a well-designed algorithm as shown in Figures 4–10 which compares NFF to TP-GAN and state-of-the-art methods in terms of visual effects. Usually, this limitation would be addressed in a face frontalization model that is designed to work with challenging datasets.

3.3. Face Identity Preservation Accuracy

The usefulness of NFF was demonstrated by comparing the classification accuracy of synthesized frontal-face images derived from the Multi-PIE and CAS-PEAL datasets. Comparisons of recognition rates are shown in Tables 3 and 4. The deep features of our data are first extracted using a light CNN model, and then their similarity is calculated using the cosine-distance metric. Using this process, we can compare our results with a baseline. As shown in Table 3, NFF preserves texture information more effectively on the Multi-PIE dataset than TP-GAN and state-of-the-art methods. Table 4 compares NFF with TP-GAN and state-of-the-art methods on the CAS-PEAL dataset, and NFF achieves better performance overall in preserving face identity information.

Large facial poses are generally associated with less facial information and structural details, leading to more performance dropouts. Most GAN-based methods fit the above description. Despite this, improving state-of-the-art methods remains a challenge and an open issue in many high-quality studies. This work addresses some of the issues associated with face recognition methods such as TP-GAN. Comparing Table 3 and Table 4, it is evident that NFF is more accurate than its competitors when addressing some of the limitations of GAN-based methods.

Table 3. Comparison of NFF recognition rates (%) with different methods using the Multi-PIE dataset. We were able to recognize most faces with rank-1 accuracy overall posture variations.

Method Face Pose	NFF	Yanfei. L., Junhua. C.	Rul. H,. et al.	Xi. Y., et al.	Chao. Y., et al.	Xiang. W., et al.	Yue. X., et al.
±90°	76.10%	67.04%	64.03%	61.02%	47.26%	5.51%	—
±75°	91.00%	87.01%	84.10%	77.02%	60.66%	24.18%	—
±60°	95.96%	93.09%	92.93%	85.02%	74.38%	62.09%	83.20%
±45°	99.56%	98.04%	98.58%	89.07%	89.02%	92.13%	86.20%
±30°	99.91%	99.07%	99.85%	92.05%	94.05%	97.38%	90.10%
±15°	99.82%	99.09%	99.78%	94.06%	96.97%	98.68%	94.00%
Average rate	93.72%	90.55%	89.87%	83.04%	77.05%	63.32%	88.37%

Table 4. A comparison of NFF recognition rates (%) across different methods using the CAS-PEAL dataset. Despite variations in posture and gray-level, we were able to recognize most faces with rank-1 accuracy.

Method Face Pose	NFF	Rul. H., et al.	Yu. Y., et al.	Jingwei. X., et al.
Pitch at (−30°)				
±15°	99.86%	98.94%	83.91%	99.72%
±30°	99.55%	98.89%	83.17%	99.56%
±45°	98.89%	97.63%	80.83%	98.99%
Average rate	99.43%	98.48%	82.63%	99.42%
Pitch at (+45°)				
±15°	98.92%	97.73%	89.44%	98.98%
±30°	99.00%	97.45%	87.95%	98.86 %
±45°	98.56%	95.83%	83.90%	98.13 %
Average rate	98.82%	97.00%	87.09%	98.65 %

Several deep learning methods have been proposed for synthesizing frontal images, but none of them are highly effective for tasks such as recognition and identification. It is stated in [18,19] that using a high-resolution face image directly from a CNN will result in sub-optimal performance. The lack of texture details in larger faces makes preserving the identity of a synthesized image more difficult. We have demonstrated that in contrast to existing methods, such as in Figures 7 and 8, the NFF exhibits impressive results at every pose level (e.g., Multi-PIE 15 to 90 degrees and CAS-PEAL 30 and 45 degrees). This may be due to the ability of NFF to generate high-quality face frontalizations, restore more facial details from any profile image, and operate under most conditions. The NFF provides us with the capabilities to build methods that are robust to facial pose challenges under a variety of datasets.

4. Results and Discussion

The face recognition system is necessary for many practical applications, including airport security, building access control, online services, etc. This type of system relies on high accuracy to improve security and streamline processes through hand-free authentication and identification. This work was motivated by the fact that pose variations are highly susceptible to facial recognition systems. Any change in face position can drastically alter facial appearance, making face recognition less accurate. The NFF provides us with valuable information on face regions, which helps us create more realistic images. TP-GAN is one of the most popular deep learning methods for generating high-quality images of frontalization faces compared to GAN-based counterparts. TP-GAN, however,

produces unsatisfactory results due to extreme face poses. To resolve the TP-GAN issue, it is important to identify and focus on a specific direction. Therefore, the NFF approach is presented to overcome the issue with TP-GAN model and improve overall performance. We would like to point out that most GAN-face frontalization methods are heavily influenced by their generator models. The training of most GAN deep learning models requires landmark detection [11,13,17,20]. An algorithm with low detection accuracy can have poor generalization abilities, while a fragile discriminator classifier architecture is capable of distinguishing between real and generated data, resulting in reduced model performance [2,5,8]. Datasets with low image resolution and insufficient samples for individuals in extreme poses lead to multi-factor failures, including the model's inability to adapt to and learn posture variations, and difficulty with training behaviors associated with different datasets [22,25,26]. NFF outperforms TP-GAN in face frontalization and state-of-the-art methods using unconstrained face recognition datasets Multi-PIE and CAS-PEAL, respectively. NFF extracts more features than other state-of-the-art methods, which allows for the better mapping and learning of any profile image of input or target poses. To demonstrate the generalization ability of NFF, we use gray-scale images in Figures 7–10. The model's abilities are tested using gray-scale directs without converting the input image to color, confirming that NFF synthesizes frontal-view faces accurately without sacrificing face identity or damaging image quality Figures 6, 8 and 10. NFF retains the facial features of the input face more efficiently, leading to more realistic frontal representations. Even with large poses, NFF can produce a good photorealistic image. We observe outstanding visual results for all the face poses using a Multi-PIE dataset in Figure 7 that contains other variations, such as low resolution and lighting variations. Several methods show severe blurring and failure cases in Figure 9, making it impossible to recover the image content when a large pose is captured due to the lack of texture details. Because of this failure, we see artifacts on the face image, which is devoid of realistic objects. NFF, on the other hand, is able to recover most of the face shapes and facial details fairly well. Others show that synthesized faces resulting from slight pose challenges (Figure 7) are beyond photorealistic. In large face pose cases, most methods fail to produce sufficiently good or reasonable results. Furthermore, these methods are not very resistant to large poses. As a result, we cannot rely on these methods to improve the final face synthesis or make it more efficient. The visual effects produced by many high-performance techniques [12,40] are inferior to ours since they diverge from the identity features of the input image. Some large poses also exhibit distortions in texture information (brightness, sharpening, clarity, and blur [50,51]). NFF produces better images with finer texture details and incorporates facial characteristics, compared to state-of-the-art methods. Our model performs better than others, which implies that it is more robust when it comes to handling large facial postures. It is important to recognize, however, that TP-GAN has three main characteristics: First, it is worth mentioning that the TP-GAN generated face image is highly dependent on local texture details and global structure, which may be influenced by posture variations, facial shapes, and other factors related to model reliability. Second, the weak classifier architecture of the TP-GAN discriminator network makes it difficult to decompose and interpret nonlinear data, which prevents it from modeling nonlinear joint distributions of images, resulting in poor identity preservation and difficulty distinguishing real data from generated data. Third, another determining factor that is found in datasets is the insufficient sample quantity for a given individual in extreme poses as well as low-quality face image resolution, causing multi-factor failures, including an inability to adapt and learn from posture variations, and difficulty in training behaviors that are associated with different datasets, which leads to poor accuracy performance. These factors may seem inconsequential, but when combined, they can degrade generalization ability and reduce

recognition accuracy. Because of this limitation, it is out of the question to perform specific facial analysis tasks, such as recognition and identification, using synthesized images. Therefore, NFF is designed to overcome these limitations, improve generalization performance under extreme face pose conditions, and maintain high levels of accuracy in face recognition. Unlike other face frontalization methods, NFF includes the above limitations by considering three components: the generator, the discriminator, and the dataset. We first assess the generator model's performance, look at posture variations and facial shapes, and develop a method that takes into account overall structure of the two counted issues. This can be accomplished through the use of additional landmark detection. Developing a classifier model that can distinguish between synthetic and real images improves face identity precision. Using decision forest, we can assist our discriminator network in achieving its objectives. Furthermore, to overcome the TP-GAN generalization abilities during the training, we must avoid two fundamental phenomena: limited face samples in extreme poses and low-quality face images during the training. We use a DA approach to increase the amount of training samples and facilitate the model's ability to learn and adapt to changes in facial features over time. Additionally, DA increases the model's ability to generalize and minimize data over-fitting. Methods such as these have the potential to produce high-level face recognition rate without significantly modifying the existing neural network models or resembling a large amount of network layers. Therefore, we study, analyze, and incorporate the above methods, and find that they provide the best solution to the issues we address throughout this paper in order to achieve effective face recognition performance accuracy.

5. Ablation Study

In this section, we will demonstrate how NFF is superior to TP-GAN. Thus, we explore different fundamental aspects (e.g., TP-GAN and NFF models) and examine their representations within two benchmark datasets. We divided our work into three sections: landmark detection, DF, and DA.

- A total of four key central facial points are selected, the eyes, the nose, and the mouth for both TP-GAN and NFF synthesized face images as shown in Table 1, while Table 5 shows the error rate for each face poses. Our landmark points are derived from a facial image using Multi-Task Cascade CNN. We use NME to measure the average distance between the profile face ID image landmark points and the synthesized face image predicted landmark points. The value of landmark detection lies in its ability to extract and integrate the most relevant features during the learning process. In landmark detection, the appearance spot of a drifted or mismatched face image is repositioned in order to recover the texture details of a synthesized face image. Figure 11 shows the central landmark-points for each face patch. For each location, we outperform the TP-GAN model: for the left eye, we reduce the error rate by 1.45%, while the right eye diminishes the error by a large margin of 2.58%; for the nose we able to shrink the error by 2.01%; and finally for the mouth, we achieve a surpass reduction result of 1.1%. Our landmark detection method expands the features extracted from the face patch components, reducing the overall error.

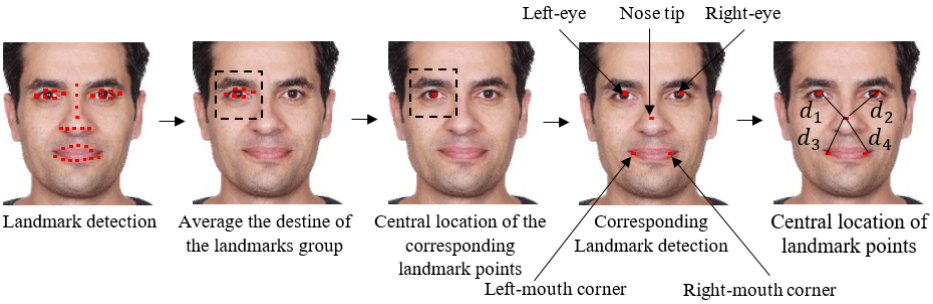


Figure 11. A breakdown of the landmark detection process that determines the landmark central points for each face patch. A central point is presented on the right-side that serves as a candidate point for evaluating the landmark algorithm.

Table 5. A comparison of error rates (ERs) (%) between the TP-GAN and NFF using the Multi-PIE dataset.

Methods Face Pose	TP-GAN ER (%)	NFF ER (%)
±90°	35.97%	23.70%
±75°	15.90%	9.22%
±60°	7.07%	4.04%
±45°	1.43%	0.46%
±30°	0.15%	0.12%
±15°	0.22%	0.13%
Average ER%	10.12%	6.61%

A challenging aspect of the landmark method is that it is not necessarily associated with high-gradient or salient points, except for a few. Based on the dataset used, landmark detection accuracy varies depending on the method. For example, Multi-PIE achieves an accuracy of 92.71%, FEI achieves an accuracy of 90.63%, and CAS-PEAL achieves an accuracy of 88.54%. Some factors cause our landmark detection to have a low detection rate, such as hair cover, sunglasses [50], and low-quality face images [51]. For example, a face with a hair cover results in a detection rate of 18.79%, while a face with sunglasses results in a detection rate of 20.71% using the Multi-PIE dataset.

- We aim here to achieve two fundamental phenomena: accurate face identification and recognizing real faces from generated ones. Although DF can be highly effective when used alone, it still has some limitations. The ability of DF to classify faces would be limited without many face samples from which to learn its decision rules. An accurate classifier model is dependent on selecting the right features and decision-making capabilities. It is therefore necessary for us to show how our DF outperforms the TP-GAN fully connected discriminator network. Experiments are conducted on the following combinations: The first part of the experiment measures the error rate of the TP-GAN discriminator network. The second part involves replacing the final fully connected TP-GAN discriminator network with DF. A light CNN model is then used to measure the accuracy of both models using Multi-PIE and FEI datasets. We compare our DF performance with stacking FC linear layers naively in the TP-GAN discriminator as shown in Table 6. In both datasets, we achieve overall impressive accuracy on Multi-PIE 2.99% and FEI 2.35%, and demonstrate that DF is better than FC TP-GAN. Further, DF is faster than FC TP-GAN in both training and testing with high-dimensional data when used with large amounts of face data.

Table 6. A comparison of the recognition accuracy (%) for Multi-PIE and FEI using FC discriminator layers (TP-GAN) and DF (our model).

Dataset Type	Method		Achieved ACC
	FC Discriminator Layer	Decision Forest (DF)	
Multi-PIE	81.24%	84.23%	2.99%
FEI	79.10%	81.45%	2.35%

- In both generalization and classification, DA has the potential to improve the TP-GAN performance. The quantities of face samples used without and with DA are provided in Table 7. DA is performed in two discrete sessions. For the first session, we use the original dataset to measure accuracy using a light CNN model. At the second sitting, we use TP-GAN to increase the number of face pose samples by generating/synthesizing new face poses from the existing dataset, followed by light CNN to determine the accuracy using the Multi-PIE dataset. A data augmentation can provide a solution to two common problems, low-quality resolution and unbalanced facial poses within datasets, which can affect curriculum learning accuracy and cause higher learning errors. For each face pose, our achieved accuracy (ACC) is displayed in the last column of Table 7, which reveals the overall ACC for each face pose. DA provides better performance with a large number of face samples, resulting in neural layer learning, more effective adoption, and enhancing the classification performance comprehensively.

Table 7. Comparison of recognition accuracy (%) without and with data augmentation using the Multi-PIE dataset.

Methods	Without Data Augmentation (%)		With Data Augmentation (%)		
Face Pose	Total Amount of Face Samples	Accuracy	Total Amount of Face Samples	Accuracy	Achieved ACC
90°	167	64.03%	269	66.14%	2.11%
75°	181	84.10%	290	86.09%	1.99%
60°	198	92.93%	279	93.56%	0.63%
45°	200	98.58%	310	98.97%	0.39%
30°	231	99.85%	300	99.96%	0.11%
15°	280	99.78%	324	99.86%	0.08%

6. Conclusions

Our paper proposes NFF for face frontalization that can produce high-quality images in extreme poses while maintaining identity information. The first step is to design and integrate landmark detection method as a way to enhance the generator's ability to generate high-quality face images. The second step is to intermingle a decision forest approach into the core architecture of the TP-GAN discriminator network to stabilize the model performance overtime. Finally, DA techniques are utilized to generate new synthetic data from existing data so that large amounts of face-pose samples can be produced while maintaining a high level of image resolution. In contrast to the existing algorithm, NFF does not require any external neural network structures, which further complicates the training process. A key feature of NFF is that it is simpler to integrate and train, as well as being more accurate and outperforming both TP-GAN and state-of-the-art methods in terms of face identification and verification. Despite the fact that deep learning has been proposed for generating frontal images, none of them have been proven to be highly

effective. The lack of information provided by large facial poses and the insufficient reliability of deep learning methods make it difficult to identify synthetic individual. We aim to demonstrate that accurate results can still be achieved even when faces are posed in extreme ways and to enhance generalization and recognition across a wide range of datasets. NFF focuses solely on improving face frontalization performance using the Multi-PIE, FEI, and CAS-PEAL datasets. Several face frontalization methods are unable to achieve high accuracy due to their inability to learn face representations. Our approach successfully recovers images from artifacts and provides high-quality face samples for the majority of postural variations. This remarkable achievement demonstrates that we are capable of reconstructing facial texture details from any input profiles and recovering missing or corrupted face parts, resulting in more photorealistic results than those generated by TP-GAN or similar face frontalization methods. Future work will focus on enhancing face realism, resolution, and diversity by improving computation efficiency and scaling up multi-machine training.

Author Contributions: M.H.B.A. conducted the experiments, analyzed the data, calculated the results, and prepared figures and tables, while Y.-S.H. revised and modified the final draft of the paper, approved the final version, and provided technical advice. All authors have read and agreed to the published version of the manuscript.

Funding: The research was not funded by any public, commercial, or non-profit sources.

Informed Consent Statement: We would like to emphasize that all datasets used in this study were open source and free to use for researchers.

Data Availability Statement: The code for this work is available at: https://github.com/MahmoodHB/A-Novel-Face-Frontalization-Method-, 1 June 2024. For Multi-PIE, the face data were downloaded from the TP-GAN GitHub repository at https://github.com/HRLTY/TP-GAN, 4 August 2017. CAS-PEAL face data were downloaded from the GitHub repository: https://github.com/YuYin1/DA-GAN, 1 February 2008. Finally, the FEI dataset was downloaded from the official repository of the FEI: https://fei.edu.br/~cet/facedatabase.html, 1 June 2005 and 1 March 2006.

Conflicts of Interest: Our paper is free from any conflicts of interest or competing financial interests that could affect or interfere with the results. As a result, we declare that no potential conflicts of interest exist.

References

1. Elharrouss, O.; Almaadeed, N.; Al-Maadeed, S.; Khelifi, F. Pose-invariant face recognition with multitask cascade networks. *Neural Comput. Appl.* **2022**, *34*, 6039–6052. [CrossRef]
2. Chatterjee, S.; Hazra, D.; Byun, Y.-C.; Kim, Y.-W. Enhancement of Image Classification Using Transfer Learning and GAN-Based Synthetic Data Augmentation. *Mathematics* **2022**, *10*, 1541. [CrossRef]
3. Chan, E.R.; Lin, C.Z.; Chan, M.A.; Nagano, K.; Pan, B.; Mello, S.D.; Gallo, O.; Guibas, L.; Tremblay, J.; Khamis, S.; et al. Efficient Geometry-aware 3D Generative Adversarial Networks. *arXiv* **2022**, arXiv:2112.07945. [CrossRef]
4. Yang, C.; Wang, Z.; Mao, S. RFPose-GAN: Data Augmentation for RFID based 3D Human Pose Tracking. In Proceedings of the IEEE 12th International Conference on RFID Technology and Applications (RFID-TA), Cagliari, Italy, 12–14 September 2022; pp. 1–4. [CrossRef]
5. Duan, Q.; Zhang, L.; Zhang, Y.; Gao, X. PSGAN: Revisit the binary discriminator and an alternative for face frontalization. *Neurocomputing* **2022**, *518*, 360–372. [CrossRef]
6. Ju, Y.-J.; Lee, G.-H.; Hong, J.-H.; Lee, S.-W. Complete Face Recovery GAN: Unsupervised Joint Face Rotation and De-Occlusion from a Single-View Image. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 3–8 January 2022; pp. 1173–1183. [CrossRef]
7. Liao, J.; Sanchez, V.; Guha, T. Self-Supervised Frontalization and Rotation Gan with Random Swap for Pose-Invariant Face Recognition. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Bordeaux, France, 16–19 October 2022; pp. 1–5. [CrossRef]

8. Yu, S.; Han, H.; Shan, S.; Chen, X. CMOS-GAN: Semi-Supervised Generative Adversarial Model for Cross-Modality Face Image Synthesis. *IEEE Trans. Image Process.* **2022**, *32*, 144–158. [CrossRef] [PubMed]
9. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein GAN. *arXiv* **2017**, arXiv:1701.07875. [CrossRef]
10. Mao, X.; Li, Q.; Xie, H.; Lau, R.Y.K.; Wang, Z.; Smolley, S.P. Least Squares Generative Adversarial Networks. *arXiv* **2016**, arXiv:1611.04076. [CrossRef]
11. Huang, R.; Zhang, S.; Li, T.; He, R. Beyond Face Rotation: Global and Local Perception GAN for Photorealistic and Identity Preserving Frontal View Synthesis. *arXiv* **2017**, arXiv:1704.04086. [CrossRef]
12. Choi, W.; Nam, G.P.; Cho, J.; Kim, I.-J.; Ko, H.-S. Integrating Pretrained Encoders for Generalized Face Frontalization. *IEEE Access* **2024**, *12*, 43530–43539. [CrossRef]
13. Jang, S.-S.; Jang, S.-S.; Kim, C.-J.; Kim, C.-J.; Hwang, S.-Y.; Hwang, S.-Y.; Lee, M.-J.; Lee, M.-J.; Ha, Y.-G.; Ha, Y.-G. L-GAN: Landmark-based generative adversarial network for efficient face de-identification. *J. Supercomput.* **2022**, *79*, 7132–7159. [CrossRef]
14. Taherkhani, F.; Talreja, V.; Dawson, J.; Valenti, M.C.; Nasrabadi, N.M. Profile to frontal face recognition in the wild using coupled conditional generative adversarial network. *IET Biom.* **2022**, *11*, 260–276. [CrossRef]
15. Zhang, Z.; Chen, Y.; Yang, W.; Wang, G.; Liao, Q. Pose-Invariant Face Recognition via Adaptive Angular Distillation. *Proc. AAAI Conf. Artif. Intell.* **2022**, *36*, 3390–3398. [CrossRef]
16. Kammoun, A.; Slama, R.; Tabia, H.; Ouni, T.; Abid, M. Generative Adversarial Networks for Face Generation: A Survey. *ACM Comput. Surv.* **2022**, *55*, 1–37. [CrossRef]
17. Xi, Y.; Jia, W.; Zheng, J.; Fan, X.; Xie, Y.; Ren, J.; He, X. DRL-GAN: Dual-Stream Representation Learning GAN for Low-Resolution Image Classification in UAV Applications. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.* **2020**, *14*, 1705–1716. [CrossRef]
18. Yin, X.; Yu, X.; Sohn, K.; Liu, X.; Chandraker, M. Towards Large-Pose Face Frontalization in the Wild. *arXiv* **2017**, arXiv:1704.06244. [CrossRef]
19. Liu, Y.; Chen, J. Unsupervised face Frontalization for pose-invariant face recognition. *Image Vis. Comput.* **2021**, *106*, 104093. [CrossRef]
20. Yin, Y.; Jiang, S.; Robinson, J.P.; Fu, Y. Dual-Attention GAN for Large-Pose Face Frontalization. *arXiv* **2020**, arXiv:2002.07227. [CrossRef]
21. Karras, T.; Laine, S.; Aila, T. A Style-Based Generator Architecture for Generative Adversarial Networks. *arXiv* **2018**, arXiv.1812.04948. [CrossRef]
22. Bulat, A.; Tzimiropoulos, G. How far are we from solving the 2D and 3D Face Alignment problem? (And a dataset of 230,000 3D facial landmarks). *arXiv* **2017**, arXiv:1703.07332. [CrossRef]
23. Sun, P.; Li, Y.; Qi, H.; Lyu, S. LandmarkGAN: Synthesizing Faces from Landmarks. *arXiv* **2022**, arXiv:2011.00269. [CrossRef]
24. Hsu, C.; Lin, C.; Hung, T.; Lei, C.; Chen, K. A Detailed Look At CNN-based Approaches in Facial Landmark Detection. *arXiv* **2020**, arXiv:2005.08649. [CrossRef]
25. Li, H.; Guo, Z.; Rhee, S.; Han, S.; Han, J. Towards Accurate Facial Landmark Detection via Cascaded Transformers. *arXiv* **2022**, arXiv:2208.10808. [CrossRef]
26. Hu, C.; Xie, X.; Wu, L. Face reenactment via generative landmark guidance. *Image Vis. Comput.* **2022**, *130*, 104611. [CrossRef]
27. Malakshan, S.R.; Saadabadi, M.S.E.; Mostofa, M.; Soleymani, S.; Nasrabadi, N.M. Joint Super-Resolution and Head Pose Estimation for Extreme Low-Resolution Faces. *IEEE Access* **2023**, *11*, 11238–11253. [CrossRef]
28. Ciuntu, V.; Chiornita, T.; Boiangiu, C.-A.; Tarba, N.; Voncila, M.-L.; Rosner, D. Image Enhancement for Facial Landmark Detection. In Proceedings of the 21st RoEduNet Conference: Networking in Education and Research (RoEduNet), Sovata, Romania, 15–16 September 2022; pp. 1–6. [CrossRef]
29. Zhang, K.; Zhang, Z.; Li, Z.; Qiao, Y. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks. *IEEE Signal Process. Lett.* **2016**, *23*, 1499–1503. [CrossRef]
30. Peter, K.; Madalina, F.; Antonio, C.; Samuel, R. Deep Neural Decision Forests. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1467–1475. [CrossRef]
31. Zuo, Y.; Avraham, G.; Drummond, T. Improved Training of Generative Adversarial Networks Using Decision Forests. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 5–9 January 2021; pp. 3492–3501. [CrossRef]
32. Yani, I.; Duncan, R.; Darko, Z.; Peter, K.; Jamie, S.; Matthew, B.; Antonio, C. Decision Forests, Convolutional Networks and the Models in-Between. *arXiv* **2016**, arXiv:1603.01250. [CrossRef]
33. Correia, A.H.C.; Peharz, R.; de Campos, C. Joints in Random Forests. In Proceedings of the Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 6–12 December 2020; pp. 1–12. [CrossRef]
34. Ruzainie, M.R.N.A.; Xiao, Y.; Uno, T.; Ma, L.; Khorasani, K. A New Facial Expression Recognition System Using Decision Tree and Deep Neural Networks. In Proceedings of the IEEE 6th International Conference on Signal and Image Processing (ICSIP), Nanjing, China, 22–24 October 2021; pp. 49–54. [CrossRef]

35. Jiang, S.; Mao, H.; Ding, Z.; Fu, Y. Deep Decision Tree Transfer Boosting. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *31*, 383–395. [CrossRef]
36. Tran, N.-T.; Tran, V.-H.; Nguyen, N.-B.; Nguyen, T.-K.; Cheung, N.-M. On Data Augmentation for GAN Training. *IEEE Trans. Image Process.* **2021**, *30*, 1882–1897. [CrossRef]
37. Liu, T.; Fan, J.; Luo, Y.; Tang, N.; Li, G.; Du, X. Adaptive data augmentation for supervised learning over missing data. *Proc. Vldb Endow.* **2021**, *14*, 1202–1214. [CrossRef]
38. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks. *arXiv* **2014**, arXiv:1406.2661. [CrossRef]
39. Yang, Y.; Morillo, I.G.; Hospedales, T.M. Deep Neural Decision Tree. *arXiv* **2018**, arXiv:1806.06988. [CrossRef]
40. Bulo, S.R.; Kontschieder, P. Neural decision forests for semantic image labelling. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1–7. [CrossRef]
41. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958. [CrossRef]
42. Murthy, S.K.; Kasif, S.; Salzberg, S. A System for Induction of Oblique Decision Trees. *J. Artif. Intell. Res.* **1994**, *2*, 1–32. [CrossRef]
43. Gross, R.; Matthews, I.; Cohn, J.; Kanade, T.; Baker, S. Multi-PIE. *Image Vis. Comput.* **2010**, *5*, 1–23. [CrossRef] [PubMed]
44. FEI. Artificial Intelligence Laboratory of FEISão Bernardo do Campo, Brazildoihttps. (2005–2006). Available online: https://fei.edu.br/~cet/facedatabase.html (accessed on 1 January 2020).
45. Gao, W.; Cao, B.; Shan, S.; Chen, X.; Zhou, D.; Zhang, X. The CAS-PEAL large-scale Chinese face database and baseline evaluations. *IEEE Trans. Syst. Man Cybern. Part A Syst. Humans* **2009**, *38*, 149–161. [CrossRef]
46. Hassner, T.; Harel, S.; Paz, E.; Enbar, R. Effective Face Frontalization in Unconstrained Images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–10. [CrossRef]
47. Xin, J.; Wei, Z.; Wang, N.; Li, J.; Wang, X.; Gao, X. Learning a High Fidelity Identity Representation for Face Frontalization. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *33*, 6952–6964. [CrossRef]
48. Lin, X.; Sun, S.; Huang, W.; Sheng, B.; Li, P.; Feng, D.D. EAPT: Efficient Attention Pyramid Transformer for Image Processing. *IEEE Trans. Multimedia* **2021**, *25*, 50–61. [CrossRef]
49. Cao, J.; Chen, Z.; Zhang, Y.; Sun, L.; Chen, J. Face frontalization with deep GAN via multi-attention mechanism. *Signal Image Video Process.* **2023**, *17*, 1965–1973. [CrossRef]
50. Liu, Y.; Chen, J.; Li, Y.; Wu, T.; Wen, H. Joint face normalization and representation learning for face recognition. *Pattern Anal. Appl.* **2024**, *27*, 64. [CrossRef]
51. Lu, Y.; Cao, J.; He, S.; Guo, J.; Zhou, Q.; Dai, Q. Learning invariant and uniformly distributed feature space for multi-view generation. *Inf. Fusion* **2023**, *93*, 383–395. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.