Face Frontalization Using Appearance Flow based Convolutional Neural Network

Zhihong Zhang, Xu Chen, Beizhan Wang*, Guosheng Hu, Wangmeng Zuo, Edwin R. Hancock, *Fellow, IEEE*

***Abstract*—Facial pose variation is one of the major factors making face recognition (FR) a challenging task. One popular solution is to convert non-frontal faces to frontal ones on which face recognition is performed. Rotating faces causes the facial pixel value changes. Therefore, existing CNN-based methods learn to synthesize frontal faces in color space. However, this learning problem in color space is highly non-linear, causing the synthetic frontal faces to lose fine facial textures. In this work, we take the view that the nonfrontal-frontal pixel changes are essentially caused by geometric transformations (rotation, translation, etc) in space. Therefore, we aim to learn the nonfrontal-frontal facial conversion in spatial domain rather than the color domain to ease the learning task. To this end, we propose an *Appearance Flow based Face Frontalization Convolutional Neural Network (A3F-CNN)*. Specifically, A3F-CNN learns to establish the dense correspondence between the non-frontal and frontal faces. Once the correspondence is built, frontal faces are synthesized by explicitly 'moving' pixels from the non-frontal one. In this way, the synthetic frontal faces can preserve fine facial textures. To improve the convergence of training, an appearance flow guided learning strategy is proposed. In addition, GAN loss is applied to achieve a more photorealistic face and a face mirroring method is introduced to handle the self-occlusion problem. Extensive experiments are conducted on face synthesis and pose invariant face recognition. Results show that our method can synthesize more photorealistic faces than existing methods in both controlled and uncontrolled lighting environments. Moreover, we achieve very competitive face recognition performance on the Multi-PIE, LFW and IJB-A databases.**

***Index Terms*—Face frontalization, Face synthesis, Optical flow, Face recognition**

I. Introduction

FACE recognition (FR) is a topical research direction in computer vision. Recently, great progress has been achieved in face recognition using deep learning methods and large database of labeled face images. However, face recognition is still a challenging problem in uncontrolled lighting environments, and in particular, in the presence of large pose variations. Specifically, strong pose variations significantly decrease the accuracy of the evaluated methods. As verified by [1], [2], [3], [4], [5], [6], pose is a major factor for reducing the accuracy. As a result, Pose Invariant Face Recognition (PIFR) has attracted great interest. Research into PIFR can be categorised into two groups a) Latent Space Learning (LSL) and b) Analysis-by-Synthesis (AbS). LSL methods are essentially general metric learning techniques from computer vision. During training, LSL methods project the features extracted from input images under various poses into a common space [7], [8] where the image features of the same identity are clustered but otherwise are far away from one another. During testing, the test face features are mapped to the same latent space for recognition. The features can be hand-crafted or learned. Hand-crafted features (SIFT [9], HOG [10], Gabor [11], LBP [12], etc.) aim to capture pose-invariant information, but the performance is not that promising. Learning-based methods, mainly deep learning methods [13], [14], [15], [16], [17], can achieve more robust PIFR performance across different poses. Hand-crafted features behave like the features from shallow layers of deep learning, which can perform low-level of robustness. Deeper layers can capture more abstract and robust information across different poses, which hand-crafted features cannot. Although LSL methods achieve promising performance, PIFR is conducted in a latent space, which is like a black box and makes the intermediate representation less interpretable. In the real world, the interpretability or visualization of the recognition process is important in many practical applications, such as law enforcement and visually identifying suspects.

To solve the interpretability problem of LSL, in contrast, AbS methods explicitly convert a face under arbitrary pose to a canonical view (frontal face) as the intermediate representation. Then face recognition can be performed with the canonical view. In this way, it is clear that the pose problem is solved by explicit frontalization, which is more interpretable. This frontalization process is also called pose normalization. AbS methods can be categorised as a) 3D methods (AbS-3D) [18], [19], [20], [21], [22], [23], [24], [25], [26] and b) 2D methods (AbS-2D) [27], [28], [29], [30], [31]. AbS-3D methods fit a 3D model, typically a 3D Morphable Model (3DMM) [32], to an input face image with arbitrary pose. After fitting, the parameters of shape, texture, pose (camera) and illumination can be recovered. By re-setting the pose parameters and keeping other parameters fixed, the input face can be re-rendered in a frontal view. Although AbS-3D methods can intrinsically handle pose transformations, the fitting process is usually slow and the performance is highly dependent on the accuracy of facial landmark detection.

Unlike AbS-3D methods, AbS-2D methods perform the frontalization in 2D space without using 3D templates (mod-

Zhihong Zhang, Xu Chen and Beizhan Wang are with the Xiamen University, Xiamen, China.

Guosheng Hu is with the Anyvision group, Belfast, UK.

Wangmeng Zuo is with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China.

Edwin R. Hancock is with the the Department of Computer Science, The University of York, York, UK.

* Corresponding author: Beizhan Wang (E-mail: wangbz@xmu.edu.cn.)

This work is supported by National Natural Science Foundation of China (Grant No.61402389) and the Fundamental Research Funds for the Central Universities in China (no. 20720160073). The first two authors contribute equally to this work.

els). AbS-2D methods can be either (1) linear or (2) non-linear. For linear methods, the spatial projection from non-frontal face to a frontal one is achieved by linear mappings [33], [34], [35]. For non-linear methods, the projection is usually achieved by deep learning methods such as CNN (convolutional neural network). In fact, CNN methods frequently use non-frontal and frontal face pairs to learn a non-linear projection by training an encoder-decoder architecture [36], [29], [31], [30]. Specifically, the non-frontal faces are usually first fed into a stack of convolution layers (encoder) with decreasing resolutions to generate compact latent representations (codes). Then the codes are passed through a stack of deconvolution layers (decoder, usually symmetrical to the encoder) to generate the desired frontal face images. These CNN-based methods achieve better performance than the linear methods. However, they generally encode the input image into pooled representation. The reconstructed frontal face is then obtained by decoding the pooled representation (bottleneck), leading to detail-losing and blurry results [29], [31], [30]. Even adversarial loss can be introduced to improve visual quality, it cannot ultimately address the blurry reconstruction issue.

To solve the blurry reconstruction problem, recently, the novel flow-based image synthesis approaches have attracted considerable attention [37], [38], [39]. The key idea underpinning these methods is to synthesize the desired image by 'moving' pixels from single or multiple input images instead of synthesizing them. For example, [37] proposed a style transfer algorithm by establishing dense correspondences between the input and the sample. In [38], CNN is trained to explicitly infer the appearance correlation between different views of objects.

Inspired by these flow-based methods, in this paper, we propose an *Appearance Flow based Face Frontalization Convolutional Neural Network (A3F-CNN)*, which aims to perform face frontalization by learning the dense correspondence between the non-frontal and frontal face images. Once such correspondence is built or learned, the frontal face image can be naturally synthesized by moving pixels from its non-frontal counterpart. To make the generated face more photorealistic, we adopt a Generative Adversarial Network (GAN) to constrain the recovery process by incorporating prior knowledge of the distribution of frontal faces. Since pose transformation is highly non-linear, we propose an appearance flow guided learning strategy. Specifically, we first apply the SIFT-FLOW algorithm [40] to establish coarse correspondences between non-frontal and frontal faces offline. Then these prebuilt correspondences are used to guide the training of A3F-CNN, making it converge quickly to desired solution. By 'moving' pixels rather than 'synthesizing' them, A3F-CNN can generate frontal faces with much richer details than CNN-based pixel synthesis methods [36], [29], [31], [30]. In addition, one of the major problems in face frontalization is to recover the self-occluded areas of a face, particularly, in the presence of large pose variations. Clearly, the perfect recovery of pixels in the self-occluded area from a single non-frontal face is intractable since this information is irreversibly lost. Fortunately, these pixels can be 'estimated' or 'guessed' by invoking facial symmetry. In fact, A3F-CNN handle large pose variations by only synthesizing the visible half part of face, and then generating the full face by concatenating the visible half face with its mirror image.

Our contributions can be summarized as:

- A novel appearance flow based end-to-end face frontalization network, A3F-CNN, is proposed. Unlike existing methods [36], [29], [31], [30], which perform frontalization learning in a color space, we perform learning in the spatial domain. Instead of reconstructing frontal faces by 'synthesizing' pixels from a black box [36], [29], [31], [30], A3F-CNN generates the desired frontal face by 'moving' pixels from their positions in a non-frontal face to those in a frontal pose. In this way, A3F-CNN can effectively preserve facial texture details. In addition, a GAN is adopted to produce photorealistic face images and facial symmetric information is used to solve the self-occlusion problem.
- To effectively train A3F-CNN, we propose an appearance flow guided learning strategy. Specifically, we first apply the SIFT-FLOW algorithm offline to establish coarse correspondences between non-frontal and frontal face images. Then the prebuilt correspondences are used to guide the training.
- A3F-CNN essentially learns the underlying spatial transformation on the 2D plane without accurately detected landmarks which are needed by AbS-3D methods.

II. Related Work

Face Frontalization, or frontal view synthesis, aims to synthesize a frontal face from a face image with arbitrary pose variation. Many methods have been proposed to solve the frontalization problem. For example, Sagonas et al. [41] propose a constrained low-rank minimization model to jointly reconstruct the frontal face and localize landmarks. Hassner et al. [18] effectively employ a shared reference 3D face model for face frontalization. Recently, many researchers have proposed Convolutional Neural Network (CNN) based methods [14], [15], [13], [36], [28], [29] for joint face frontalization and representation learning, and they have achieved impressive improvement in performance. For instance, Kan et al. [36] progressively rotate a non-frontal face image to a frontal one through multiple stacked auto-encoders. Yin et al. [30] propose FF-GAN, where 3DMM was incorporated into the GAN architecture to provide shape and appearance priors. Huang et al. [31] use a two-pathway GAN architecture for simultaneously perceiving global structures and local details. Tran et al. [29] propose an encoder-decoder network, named DR-GAN, to simultaneously learn a pose-invariant face representation and synthesize the frontal face. Specifically, DR-GAN can take multiple images as the inputs to synthesize a frontal face, which differentiates it from its counterparts.

Dense Correspondence, is a non-trivial problem, and aims to establish pixel-level correspondence across images. Most typically it operates with two images. Previously, researchers construct the correspondences between two images under the brightness constancy assumption [42], [43]. Unfortunately, this has been proven to be vulnerable to variations caused by

lighting, perspective and noise [44]. Middle-level features, such as SIFT [9] and HOG [10], can be used as a more robust image representations, and achieve great success in many applications [45], [46], [47]. For example, Liu et al. propose SIFT-FLOW [40] which aligns two images from different 3D scenes by pixel-wise matching SIFT features between them. Most recently, a variety of techniques [37], [38], [39], [48] aim to estimate correspondence between pair or multiple images using CNNs, which can learn more robust features than hand-crafted ones such as SIFT, HOG, etc.

Learning Geometric Transformations for warping generally can result in image with fine details. In [49], spatial transformer networks (STNs) are suggested to learn a spatial mapping. Subsequently, CNN models are further exploited for dense flow estimation [38], [39], [50]. These methods generally adopt an encoder-decoder architecture to predict dense flow. The flow networks are then learned by minimizing the pixel loss enforced on the warped and the ground-truth images. However, due to that face image usually is smooth, the flow network by pixel loss intends to be trapped into undesired local minima for face frontalization. In this work, in addition to pixel loss, we further incorporate dense correspondence loss by middle-level features to address this issue.

Generative Adversarial Network (GAN), introduced by Goodfellow et al. [51], has recently attracted attention in the field of deep learning. The key idea underpinning GAN is to train two networks, i.e. a generator and a discriminator, in turn in an adversarial way. Specifically, the generator is trained to produce a synthetic photorealistic image to fool the discriminator, while the discriminator learns to distinguish the synthetic image from the real one. With a mini-max two-player game, the generator and discriminator compete with each other and can mutually improve performance. Since GAN is able to generate photorealistic images with plausible high frequency details, it is used in a wide range of applications, such as image generation [52], [53], super-resolution [54], style transfer [55], and face hallucination [56].

III. Methodology

Assume that we have a pair of images (I^P, I^F), where I^P and I^F represent two face images with same identity but viewed from non-frontal and frontal directions, respectively. For simplicity, we also assume both images are of the same size of $H \times W \times C$ with H the height, W the width, C the number of color channels. In this work, we aim to synthesize a frontal face $\hat{I}^F$ with rich facial texture details from a non-frontal face I^P. At the same time, the identity information should also be well preserved.

In this section, we propose *Appearance Flow based Face Frontalization Convolutional Neural Network (A3F-CNN)*, which incorporates flow based dense correspondence into deep learning based frontalization. Specifically, we reconstruct the frontal face by 'moving' pixels from the input non-frontal face to the target one, rather than 'generating' pixels on the fly. The pixel movement can guarantee that the synthetic frontal faces contain fine details. This pixel movement is achieved by a network, Generator, which learns the pixel-wise spatial transformation between the non-frontal face and the frontal one. Additionally, adversarial loss is adopted to generate photorealistic output. The architecture of the network is detailed in Section III-A. This frontalization operation ('pixel movement' process) is highly non-linear, causing difficulties with network training. In this work, we propose an appearance flow guided learning strategy to alleviate the training difficulty, and which is detailed in Section III-B. Subsequently, to solve the self-occlusion problem in face frontalization, a symmetry based face mirroring method is introduced in Section III-C. Finally, the synthesis loss function of our method is detailed in Section III-D. The general framework of the proposed method is shown in Fig. 1.

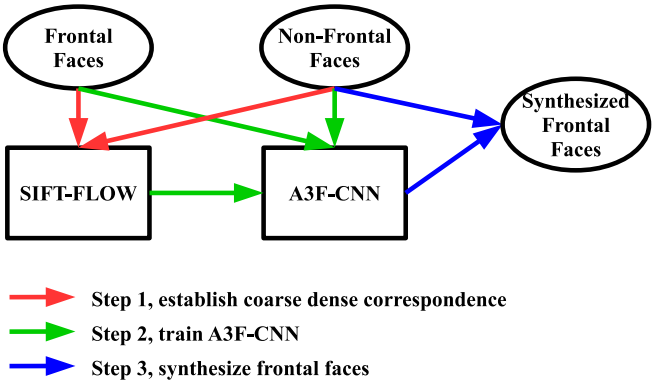


Fig. 1. The general framework of the proposed method. First, coarse dense correspondences between non-frontal and frontal faces are offline generated by using SIFT-FLOW algorithm. Then, these pre-computed coarse dense correspondences can be used to guide the training of A3F-CNN.

A. Network Architecture

The *generator* of A3F-CNN is illustrated in Fig. 2 where the output size of each block is labeled. It comprises a) an encoder, b) a decoder and c) a sampling operator. Specifically, the encoder takes a non-frontal face image as input, followed by several encoder blocks. Each encoder block consists of a strided convolution layer that reduces the spatial size and a residual block [57] that has strong non-linear learning capacity. The decoder takes the output of encoder as input, followed by several decoder blocks, ending with a convolution layer that generates the sampling coordinates. Similarly, each decoder block consists of a deconvolution layer [58] that magnifies the size of feature maps and a residual block. Finally, the frontal face image is generated by sampling pixels from the input using a bilinear sampling method according to the estimated sampling coordinates. We adopt $PRelu$ [59] as the activation function for each convolution/deconvolution layer except for the final one, where $tanh$ is applied to normalize the output (image pixel coordinates). Batch normalization [60] is also used after each convolution/deconvolution layer except the first one. The architecture of our generator is detailed in Table I.

Note that the generator consists of three components: an encoder, a decoder and a bilinear sampling operation. Given the input I^P, we denote by $G(I^P)$ the output of whole generator, i.e., the synthesized frontal face. And $C(I^P)$ denotes the output of decoder, which is actually the predicted sampling

TABLE I
THE DETAILED STRUCTURE OF GENERATOR. RB REPRESENTS RESIDUAL BLOCK [57].

Encoder		Decoder	
Layer	Filter/Strides	Layer	Filter/Strides
Conv0	$5 \times 5/1$	DeConv3	$5 \times 5/3$
Conv1	$3 \times 3/2$	RB5	$3 \times 3/1$
RB1	$3 \times 3/1$	DeConv2	$3 \times 3/2$
Conv2	$3 \times 3/2$	RB6	$3 \times 3/1$
RB2	$3 \times 3/1$	DeConv1	$3 \times 3/2$
Conv3	$3 \times 3/2$	RB7	$3 \times 3/1$
RB3	$3 \times 3/1$	DeConv0	$3 \times 3/2$
Conv4	$5 \times 5/3$	RB8	$3 \times 3/1$
RB4	$3 \times 3/1$	Conv5	$5 \times 5/1$

coordinates. Besides, $C_i(I^P)$ represents the value of $C(I^P)$ at position i. The bilinear sampling operation, introduced by [49], has the form

$$\begin{aligned} BS_i(I^P, C_i(I^P)) = \\ \sum_{j \in N(C_i(I^P))} I^P_j max(0, 1 - |C^x_i(I^P) - x_j|) \\ max(0, 1 - |C^y_i(I^P) - y_j|) \end{aligned} \tag{1}$$

where $N(C_i(I^P))$ represents the set of 4 neighbours of $C_i(I^P)$, (x_j, y_j) denotes the absolute coordinates of the pixel at position j. Note that this sampling operation is differentiable, which means that the whole network can be trained in an end-to-end manner.

To generate photorealistic faces, adversarial loss is adopted to guide the synthetic face following the target distribution of real frontal faces. The structure of the *Discriminator* is similar to CASIA-Net [61] except that Max-Pooling and Fully Connected layers are replaced with convolution layers according to [62]. In addition, batch normalization [60] is used before each convolutional layer except the first one. Leaky ReLU [63] with slope 0.2 is adopted as the activation function after each convolution layer except for the last one.

B. Appearance Flow Guided Learning Strategy

Unlike many CNN-based methods which are designed to generate the reconstruction image directly [36], [29], [31], [30], A3F-CNN only learns the spatial transformation guided by optical flow. The learning of A3F-CNN is highly non-linear and is easily trapped into local minima. Empirically, we find that the training of A3F-CNN has a high probability to fail without proper initialization. To solve this problem, we adopted an appearance flow guided learning strategy. Specifically, offline processing is first applied to learn the 'coarse' dense correspondence between input (non-frontal) and output (frontal) faces. This 'coarse' dense correspondence can guide the network training to quickly converge to a satisfying point. In this work, this correspondence is achieved by using SIFT-FLOW [40] approach in an offline fashion.

SIFT-FLOW was proposed as an image alignment method, aiming to align an image to its nearest neighbours in a large image corpus containing a variety of scenes. The SIFT-FLOW algorithm consists of two components: pixel-wise SIFT feature extraction and matching. Although the original SIFT descriptor [9] is a feature representation method consists of both feature extraction and detection, only the feature extraction component is used in SIFT-FLOW algorithm. Compare with the original optical flow methods that build pixel-level correspondence between two images, the SIFT descriptor in SIFT-FLOW can characterize local image structures and encode contextual information, which contributes to achieve a robust matching across various scene or object appearances.

The design of matching objective function of SIFT-FLOW is based on two criteria: (1) the SIFT feature should be matched along the flow vector and (2) the flow field should be smooth except on object boundaries. Let s_1 and s_2 represent two SIFT images, and denote the coordinate of image by $p = (x, y)$, denote the flow vector at position p by $w(p) = (u(p), v(p))$, the objective function of SIFT-FLOW is formulated as follow:

$$\begin{aligned} E(w) = \sum_p min(\|s_1(p) - s_2(p + w(p))\|_1, t) + \\ \sum_p \eta(|u(p)| + |v(p)|) + \\ \sum_{p,q \in \epsilon} (min(\alpha|u(p) - u(q)|, d) + \\ min(\alpha|v(p) - v(q)|, d)) \end{aligned} \tag{2}$$

where η and α are weighting parameters, t and d denote thresholds. It is clear that this objective function consists of three components: data term, small displacement term and smoothness term (or spatial regularization term). The data term constrains the SIFT features to be matched along with the flow vector $w(p)$. The small displacement term limits the size of flow vectors, and the smoothness term requires the flow vector of adjacent pixels to be similar. By optimizing this objective function, the correspondence between two images can be estimated. Fig. 3 shows two examples of dense correspondences generated by SIFT-FLOW algorithm.

In A3F-CNN, SIFT-FLOW creates the 'coarse' correspondence between input and target synthetic images. This 'coarse' correspondence is then used to guide the optimization (synthesis) process to avoid trapping into local minima. This process is detailed in Section III-D2.

Fig. 3. Dence correspondence established by SIFT-FLOW approach.

C. Symmetry Based Face Mirroring Method for Self-Occlusions

Another challenge in face frontalization is self-occlusion. In particularly, in the presence of large pose variations. This

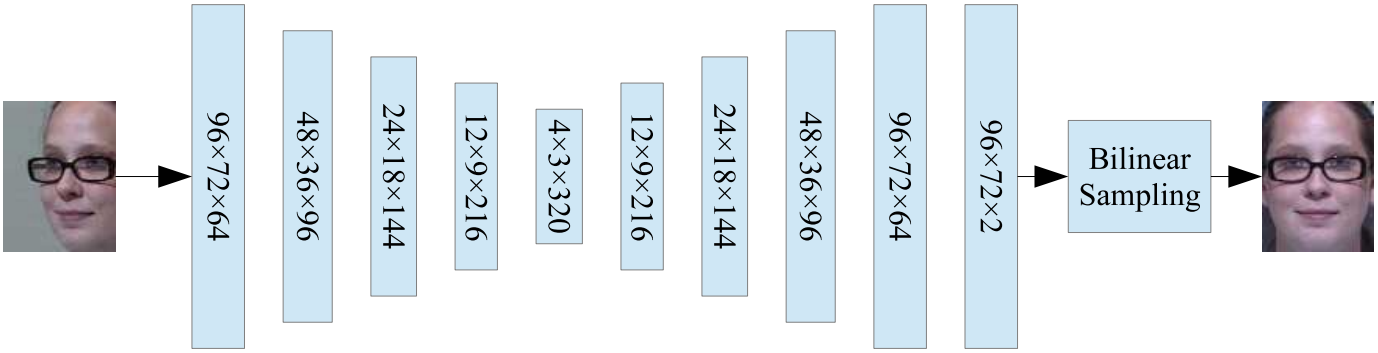


Fig. 2. General structure of generator.

problem exists in all the frontalization methods including our flow-based frontalization. Clearly, it is impossible to perfectly recover the frontal face from a non-frontal one since the information in self-occluded areas is irreversibly lost. Usually, we can only 'guess' the pixels in the occluded area from the unoccluded area based on the assumption that faces are roughly symmetrical. In this work, we adopt a face mirroring method to solve the self-occlusion problem by exploiting the facial symmetry prior [64], [65], [21], [66]. As illustrated in Fig. 4, A3F-CNN just recovers half of the frontal face, then the other half is mirrored from the unoccluded half face.

D. Loss Functions

In our work, we define the objective function as a weighted sum of 4 individual loss functions,

$$L = L_{pixel} + \lambda_1 L_{dc} + \lambda_2 L_{ip} + \lambda_3 L_{adv} \tag{3}$$

where L_{pixel}, L_{dc}, L_{ip} and L_{adv} are the pixel-wise loss, the dense correspondence loss, the identity preserving loss and the adversarial loss, respectively. The details of these 4 losses are as follows.

1) Pixel-wise Loss: As ℓ_2 loss tends to generate blurry output, we adopt the ℓ_1 loss to better preserve high frequency signals. The formulation of the pixel loss term, or fidelity term, is as follows,

$$L_{pixel} = \frac{1}{H \times W} \left\| G(I^P) - I^F \right\|_1 \tag{4}$$

Note that in case of large pose difference, the pixel-wise loss is only calculated in half of the facial image due to the other half will be recovered by the mirroring, as stated in Section III-C.

2) Dense Correspondence Loss: The coarse dense correspondences between the non-frontal and frontal faces that are generated offline by the SIFT-Flow [40] method are used to guide the training through the dense correspondence loss term.

$$L_{dc} = \frac{1}{H \times W} \left\| C(I^P) - DC(I^P, I^F) \right\|_1 \tag{5}$$

where $DC(I^P, I^F)$ indicates the prebuilt dense correspondence between I^P and I^F. This term constrains the network to learn the appearance flow from non-frontal input to frontal output guided by a dense correspondence that is generated offline. Note that we still adopt the ℓ_1 loss in order to tolerate the imprecise prebuilt correspondences.

3) Identity Preserving Loss: To preserve the identity while synthesizing frontal face, a pre-trained face recognition network is used to apply content loss [67] between the synthetic image and its ground-truth counterpart. To be more specific, features extracted from the synthetic image are required to be close to the features from ground-truth one, so as to obtain an identity preserving ability. The identity preserving loss term is as follows,

$$L_{ip} = \frac{1}{\#F} \left\| F(G(I^P)) - F(I^F) \right\|_2 \tag{6}$$

where $\#F$ represents the feature dimension, and $F(*)$ is the feature extractor of the pre-trained recognition network.

4) Adversarial Loss: In order to generate photorealistic frontal faces, we also adopt an adversarial loss, which aims at forcing the synthetic frontal face image to reside on the manifold of real frontal face images. In this work, we use Least Square GAN [68] since it is more stable than the original GAN [51]. The adversarial loss term is as follows,

$$L_{adv} = (D(G(I^P)) - c)^2 \tag{7}$$

where c is set to be 1 in our work.

IV. Experiments

In this section, we first describe the detailed experimental settings, including the face databases, hyper-parameters used in experiments. Then we present some qualitative results, i.e., the visualization of our synthetic frontal pose face image. We also quantitatively evaluate the performance of the proposed method on face recognition, demonstrating that A3F-CNN can generate highly discriminative features for face recognition. Finally, an analysis is conducted to study the effect of different components of A3F-CNN.

A. Experimental Settings

The face databases used in this experiment include:

- **Multi-PIE** [69], the largest database for evaluating face recognition under pose, illumination and expression variations in controlled environments. It contains $750,000+$

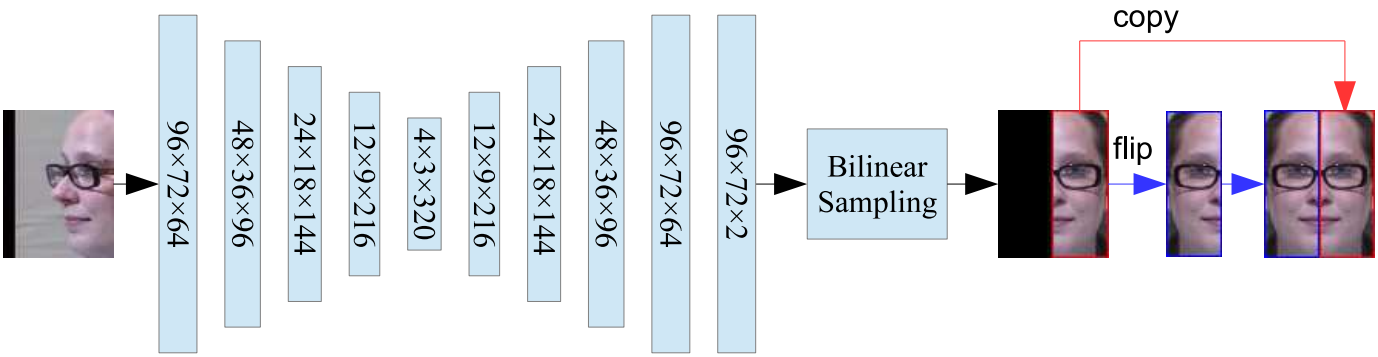


Fig. 4. Face mirror method for large pose variations.

images taken from 337 subjects, with 13 poses and 20 illuminations.

- **CASIA-WEBFACE** [61], a popular face database for training deep face models, consisting of $494,414$ images from $10,575$ subjects.
- **LFW** [70], a well-known face database for evaluating the performance of face recognition in the wild, consisting of $13,233$ images collected from the web. In its verification protocols, the test set consists of 10 folds, each with 300 matched pairs and 300 unmatched pairs.
- **IJB-A** [71], also known as **IARPA Janus Benchmark A**, is a challenging large pose face database. It has $5,396$ images and $20,412$ video frames for 500 subjects in uncontrolled settings.

The experiments consist of two parts: a) synthesis and b) recognition. For synthesis, the proposed synthesis method is trained on database (Multi-PIE), and validated on the Multi-PIE, LFW and IJB-A databases. Following the evaluation settings in [29], we use a subset of Multi-PIE with all four sessions, 337 subjects, 9 poses from $-60°$ to $60°$ and 20 illuminations. The first 200 subjects are used for training while the remaining 137 subjects are for testing. The number of images used for training and testing are $299,340$ and $153,180$, respectively. For recognition, the face recognition network, CASIA-NET, is trained using the whole CASIA-WEBFACE database and the aforementioned training set of Multi-PIE. The face recognition performance is tested on the Multi-PIE, LFW and IJB-A databases. In addition, to determine whether to use face mirroring method ($\geq 45°$), the pose of the face is first estimated using method introduced in [72].

All the images used in experiment are cropped and scaled to size of $96 \times 72 \times 3$ ($H \times W \times C$), while the pixel values are normalized into the range of $[-1, 1]$. We train our network using the *Adam* [73] optimizer with a learning rate of 10^{-4}. Other hyper-parameters are empirically set as $\lambda_1 = 0.1$, $\lambda_2 = 10^{-5}$, $\lambda_3 = 10^{-4}$, where λ_1, λ_2 and λ_3 are introduced in III-D. Our model is trained on a NVIDIA Tesla K80 GPU, but only one of two cores is used. The training lasts $100,000$ iterations, which takes around 20 hours. In the testing phrase, processing each image takes about 0.0003 second.

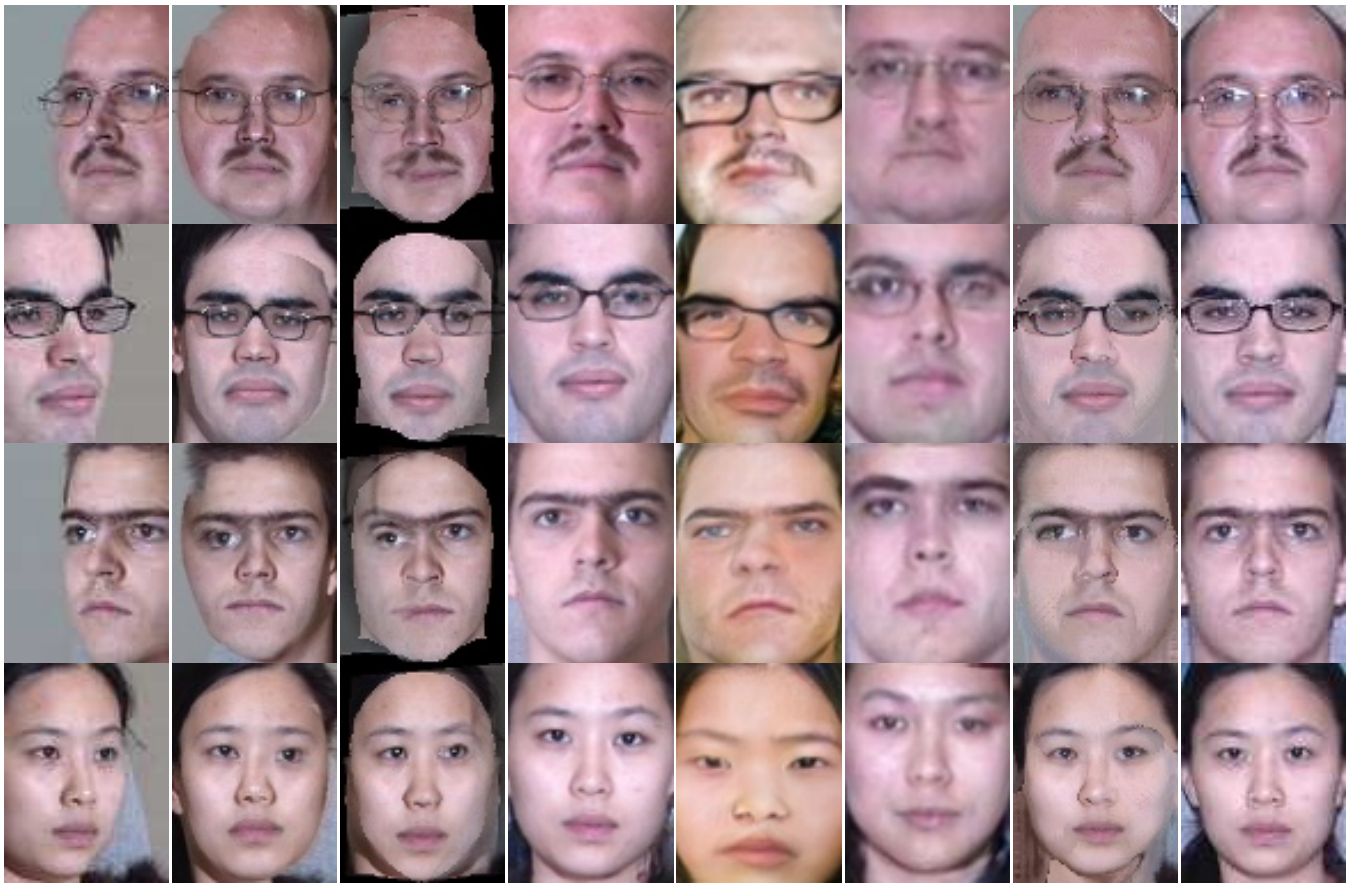
B. Qualitative Evaluation-Face Synthesis

Many face frontalization methods suffer from the problem of missing high frequency facial details even under small pose variations. As a result, they tend to generate blurry images lacking fine details. In this section, we demonstrate that A3F-CNN can generate frontal faces with rich texture details by moving pixels instead of synthesizing them. Fig. 5 shows a comparison between A3F-CNN and several state-of-the-art face frontalization methods [74], [18], [31], [29], [30], where GT represents ground-truth. It is clear that the traditional AbS-3D methods [74], [18] cannot faithfully recover the shape of the face especially in self-occluded areas. Moreover, the synthetic faces contain many strong artifacts. Not surprisingly, the images generated by most of CNN-based methods [29], [30] lack of fine facial details partially due to the bottleneck of the encoder-decoder network. In contrast, A3F-CNN can generate photorealistic synthetic face image as well as preserve texture details. Although the recently proposed TP-GAN [31] can also synthesize face images with rich textures, it requires main facial components (two eyes, nose and mouth) of the face to be accurately located, which means that the performance of landmark detection algorithm can significantly affect the quality of synthesized frontal face.

To analyze the effectiveness of our method on different poses, more results are illustrated in Fig. 6. We can see that our method can effectively recover the frontal view of face in small pose cases. While in large pose cases, since the self-occlusion problem is serious, the frontal face is concatenated by visible half part of face and its mirrors, as introduced in Section III-C. Even so, A3F-CNN can still generate photorealistic faces.

As our synthesis model is trained using the images in a controlled environment (Multi-PIE), it is interesting to know whether the trained model can generalize well to faces in an uncontrolled environment. To evaluate this generalization capacity, we test A3F-CNN on the LFW and IJB-A datasets. As shown in Fig. 7, A3F-CNN can also recover photorealistic frontal faces from faces in the wild, illustrating the strong generalization capacity of A3F-CNN.

Fig. 8 shows pixel correspondences predicted by the Generator. It is obvious that pixels of synthetic frontal face are mainly sampled from their counterparts non-frontal input facial images. In other words, A3F-CNN has essentially learned



(a) Input (b) [74] (c) [18] (d) [31] (e) [29] (f) [30] (g) Ours (h) GT

Fig. 5. Comparison with state-of-the-art face frontalization methods.

(a) 60° (b) 45° (c) 30° (d) 15° (e) -15° (f) -30° (g) -45° (h) -60°

Fig. 6. Face frontalization from arbitrary poses in constrained environment on Multi-PIE.

Fig. 7. Face frontalization from arbitrary poses in the wild on LFW (Columns 1-3) and IJB-A (Columns 4-6).

the underlying pose transformation rule between frontal and non-frontal faces. Note that A3F-CNN does not rely on any 3D knowledge; the training is conducted through data-driven learning on the 2D plane alone.

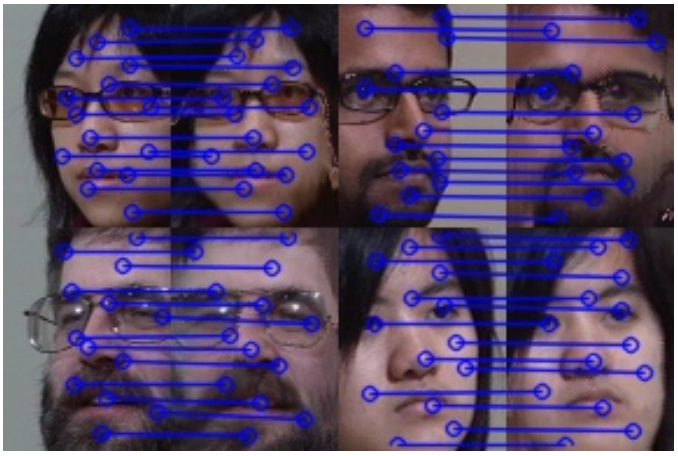


Fig. 8. Dense correspondence predicted by our method.

C. Face Recognition

To quantitatively demonstrate that our proposed method can generate identity preserving faces, we first conduct face recognition on Multi-PIE. In this experiment, we randomly select exactly one image with a frontal view for each subject in the testing set for used as a gallery, leaving the remaining as probe images. Each image is first passed into A3F-CNN to generate the corresponding frontal view. Then deep features are extracted from the generated image using a pre-trained face recognition network (CASIA-NET). Rank-1 recognition accuracy is evaluated with a cosine-distance metric. Evaluation results are shown in Table II. Here A3F-CNN achieves the best performance for all poses. The favorable performance of our method indicates that our model can synthesize more photorealistic and identify-preserving frontal faces.

In addition, we also evaluate the face recognition performance of A3F-CNN on LFW and IJB-A databases, where faces are taken from an uncontrolled environment. As shown in Table III, our method achieves the best mean face verification accuracy in comparison to its counterparts, demonstrating that it effectively preserves identity-related texture information.

TABLE II
COMPARISON OF STATE-OF-THE-ART METHODS IN TERMS OF RECOGNITION ACCURACY (%) ON MULTI-PIE.

Methods	±15°	±30°	±45°	±60°	mean
Zhu et al. [15]	90.7	80.7	64.1	45.9	70.3
Zhu et al. [13]	92.8	83.7	72.9	60.1	77.4
CFP [14]	95.0	88.5	79.9	61.9	81.3
DR-GAN [29]	94.0	90.1	86.2	83.2	88.4
FF-GAN [30]	94.8	93.4	91.0	87.0	91.5
TP-GAN [31]	**98.7**	98.1	95.4	87.7	95.0
CASIA-NET [61]	98.1	97.5	95.1	90.5	95.3
A3F-CNN	**98.7**	**98.9**	**95.8**	**92.7**	**96.5**

Moreover, Table IV shows the verification and identification performance on IJB-A database. On both verification and identification test, our method achieves consistently better results than many state-of-the-art methods.

TABLE III
FACE VERIFICATION RESULTS ON LFW.

Methods	ACC(%)	AUC(%)
Hassner et al. [18]	93.62 ± 1.17	98.38 ± 0.06
HPEN [74]	96.25 ± 0.76	99.39 ± 0.02
FF-GAN [30]	96.42 ± 0.89	$\mathbf{99.45} \pm 0.03$
CASIA-NET [61]	96.43 ± 0.97	99.29 ± 0.32
A3F-CNN	$\mathbf{96.63} \pm 0.99$	99.29 ± 0.42

TABLE IV
PERFORMANCE COMPARISON ON IJB-A DATABASE.

Methods	Verification		Identification	
	FAR=0.01	FAR=0.001	Rank-1	Rank-5
OpenBR[71]	23.6 ± 0.9	10.4 ± 1.4	24.6 ± 1.1	37.5 ± 0.8
GOTS[71]	40.6 ± 1.4	19.8 ± 0.8	44.3 ± 2.1	59.5 ± 2.0
Wang[75]	72.9 ± 3.5	51.0 ± 6.1	82.2 ± 2.3	93.1 ± 1.4
PAM[72]	73.3 ± 1.8	55.2 ± 3.2	77.1 ± 1.6	88.7 ± 0.9
DCNN[76]	78.7 ± 4.3	-	85.2 ± 1.8	93.7 ± 1.0
DR-GAN[29]	77.4 ± 2.7	53.9 ± 4.3	85.5 ± 1.5	94.7 ± 1.1
CASIA-NET	78.0 ± 1.9	57.3 ± 9.1	91.8 ± 1.9	96.1 ± 1.0
A3F-CNN	$\mathbf{80.4} \pm 3.3$	$\mathbf{60.0} \pm 8.6$	$\mathbf{92.2} \pm 2.3$	$\mathbf{97.4} \pm 0.9$

D. Ablation Study

In this section, we study the effect of the various model components used in our work, including a) the appearance flow guided learning strategy, b) the ℓ_1 pixel loss term, c) the face mirroring method for self-occlusions and d) the adversarial loss.

Appearance Flow Guided Learning Strategy: To validate the effectiveness of the proposed appearance flow guided learning strategy, we train the A3F-CNN directly without a dense correspondence loss term for comparison, and the pixel loss was monitored during training. The comparison of pixel loss with and without dense correspondence guidance is illustrated in Fig. 9, while the corresponding frontalization results are shown in Fig. 10. The learning without dense correspondence guidance obviously becomes trapped into undesired local minima, and the trained model can only roughly generate a blurry face rather than a photorealistic one. Fig. 11 shows the predicted dense correspondence by model trained without L_{dc}. It is clear that the prebuilt coarse dense correspondence is essential to assist the network to avoid being trapped in local minima.

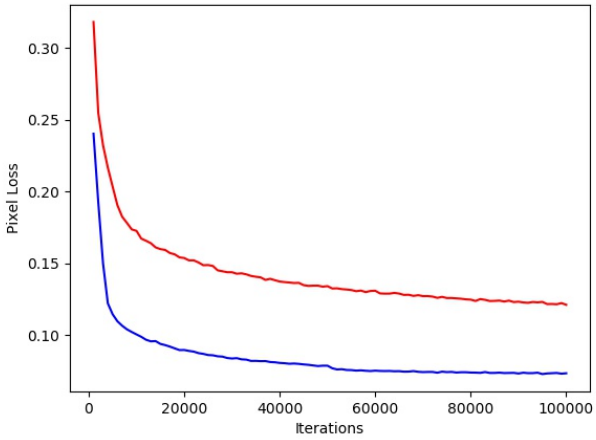


Fig. 9. The pixel loss values during training. The blue and red curves represent the pixel loss values of network trained with and without dense correspondence loss term, respectively.

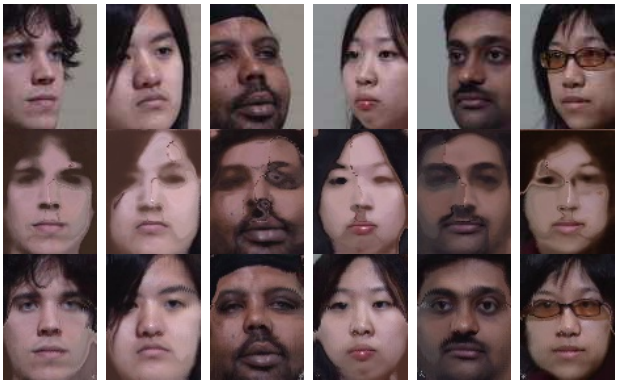


Fig. 10. Synthesis results without (middle row) and with (last row) dense correspondence constraint.

Fig. 11. Dense correspondence predicted by method trained without L_{dc}.

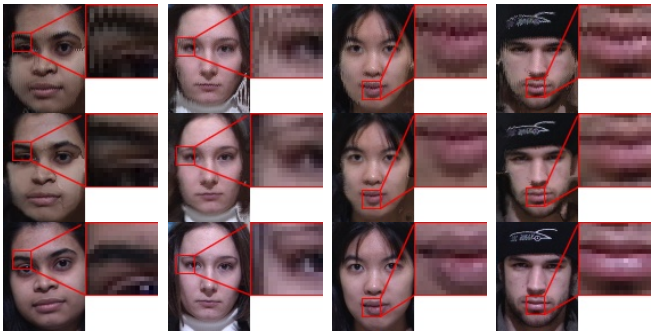


Fig. 12. Comparison of face synthesis results with $l2$ (first row) and $l1$ (second row) loss terms. While the last row represents the ground truth.

Fig. 13. Comparison of face synthesis results without (middle row) and with (last row) concatenate process. While the first row represents the input.

ℓ_1 vs. ℓ_2 Pixel-wise Loss: ℓ_1 loss and ℓ_2 loss are two of the most commonly used fidelity terms in image synthesis. As discussed in [77], ℓ_2 loss suffers from several well-known limitations in image synthesis. For example, the use of ℓ_2 loss assumes that the noise is independent of the local characteristics of the image, which is usually not valid in face synthesis. To qualitatively compare the synthetic results under ℓ_2 loss with our ℓ_1 loss, we train a model where the fidelity term is replaced by the ℓ_2 loss term, while the training strategy is kept the same for a fair comparison. As shown in Fig. 12, images generated by the ℓ_2 loss are relatively blurry and contain more visible artifacts. In contrast, the model trained using the ℓ_1 loss can better maintain high frequency details and the synthetic faces are more similar to the ground-truth ones.

Face Mirroring Method: To handle the self-occlusion problem for large pose variations, we adopt a face mirroring method that concatenates the visible half face with its mirror image. To demonstrate the effectiveness of the face mirroring method, we compare the faces generated with and without the mirroring processing under large pose variations. The results are shown in Fig. 13, where the last row represents the synthetic faces without mirroring processing. It is clear that given a single non-frontal face with large pose, the self-occluded part is rather difficult to recover. In contrast, the face mirroring method uses the visible part to recover the occluded part, generating more realistic faces (middle row).

Adversarial Loss: In order to generate more photorealistic images, we apply GAN through the adversarial loss term, aiming at forcing the synthetic faces to match the target distribution of real frontal ones. To show the effect of adversarial loss, we train the model without the adversarial loss term for comparison. As shown in Fig. 14, faces generated by the model trained without the adversarial loss term (the first row) contain more artifacts. In contrast, adversarial loss can successfully suppress artifacts, making the synthetic images (the second row) more photorealistic.

Fig. 14. Comparison of face synthesis results without (first row) and with (second row) adversarial learning. While the last row represents the ground truth.

Quantitative results: Finally, we quantitatively analyze the effect of each component. We first train A3F-CNN without certain component and then evaluate the trained model in terms of recognition accuracy on Multi-PIE. The result is shown in Table V. We can find that the model trained with ℓ_2 pixel loss instead of ℓ_1 achieves comparable performance with original A3F-CNN. The same conclusion can be drawn by model trained without adversarial learning. This suggests that the ℓ_2 pixel loss and adversarial learning have limited effect on recognition. In contrast, the performance of models trained without L_{dc} and mirroring method drop significantly. That is not surprising. As discussed above, without these two components, the training can easily trap into local minima and fail to generate realistic faces.

TABLE V
ABLATION STUDY IN TERMS OF RECOGNITION ACCURACY (%) ON MULTI-PIE.

Methods	±15°	±30°	±45°	±60°	mean
w/o L_{dc}	95.3	94.3	92.1	87.3	92.3
ℓ_2 pixel loss	98.5	98.0	96.1	92.9	96.4
w/o mirroring method	98.7	98.9	92.3	88.5	94.6
w/o L_{adv}	98.5	98.2	95.6	92.9	96.3
A3F-CNN	98.7	98.9	95.8	92.7	96.5

V. CONCLUSIONS AND DISCUSSIONS

In this work, we have proposed a face frontalization method, which we refer to as Appearance Flow based Face Frontalization Convolutional Neural Network (A3F-CNN). Instead of directly estimating pixel values as alternative CNN-based methods do, A3F-CNN is trained to learn the dense correspondence between non-frontal and frontal face images, while the desired output is synthesized by sampling pixels from the input. In addition, an appearance flow guided learning strategy

is introduced to alleviate the training problem together with a face mirroring method which is used to handle the self-occlusion problem. Compared to competing methods, A3F-CNN can generate frontal faces with rich texture details as well as preserve identity information. However, A3F-CNN has relatively weak capacity for face frontalization with extreme poses (i.e., profile faces). In this case the facial textures in many major facial regions (e.g., eyes, mouth, etc.) are totally different in profile and frontal face images. Another weakness of A3F-CNN is that the boundary between face and background in the synthetic images tend to be blurred. This is because the correspondence of pixels around the boundary can vary widely, which confounds the recovery process. Despite these, A3F-CNN can still be seen as a powerful face frontalization method due to its detail preserving capability.

REFERENCES

[1] S. Sengupta, J.-C. Chen, C. D. Castillo, V. M. Patel, R. Chellappa, D. W. Jacobs, Frontal to profile face verification in the wild, in: Workshop on Applications of Computer Vision, 2016, pp. 1–9.

[2] M. Kan, S. Shan, X. Chen, Multi-view deep network for cross-view classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4847–4855.

[3] X. Chai, S. Shan, X. Chen, W. Gao, Locally linear regression for pose-invariant face recognition, IEEE Transactions on Image Processing 16 (7) (2007) 1716–1725.

[4] C. Ding, D. Tao, A comprehensive survey on pose-invariant face recognition, ACM Transactions on intelligent systems and technology 7 (3) (2016) 37.

[5] X. Liu, T. Chen, Pose-robust face recognition using geometry assisted probabilistic modeling, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2005, pp. 502–509.

[6] X. Liu, J. Rittscher, T. Chen, Optimal pose for face recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2006, pp. 1439–1446.

[7] D. Chen, X. Cao, F. Wen, J. Sun, Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 3025–3032.

[8] K. Q. Weinberger, L. K. Saul, Distance metric learning for large margin nearest neighbor classification, Journal of Machine Learning Research 10 (Feb) (2009) 207–244.

[9] D. G. Lowe, Object recognition from local scale-invariant features, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, 1999, pp. 1150–1150.

[10] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2005, pp. 886–893.

[11] J. G. Daugman, Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters, Journal of the Optical Society of America A Optics & Image Science 2 (7) (1985) 1160–9.

[12] T. Ahonen, A. Hadid, M. Pietikainen, Face description with local binary patterns: Application to face recognition, IEEE Transactions on Pattern Analysis & Machine Intelligence 28 (12) (2006) 2037–2041.

[13] Z. Zhu, P. Luo, X. Wang, X. Tang, Multi-view perceptron: a deep model for learning face identity and view representations, in: Advances in Neural Information Processing Systems, 2014, pp. 217–225.

[14] J. Yim, H. Jung, B. Yoo, C. Choi, D. Park, J. Kim, Rotating your face using multi-task deep neural network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 676–684.

[15] Z. Zhu, P. Luo, X. Wang, X. Tang, Deep learning identity-preserving face space, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 113–120.

[16] X. Peng, X. Yu, K. Sohn, D. N. Metaxas, M. Chandraker, Reconstruction-based disentanglement for pose-invariant face recognition, in: IEEE International Conference on Computer Vision, 2017, pp. 1632–1641.

[17] W. AbdAlmageed, Y. Wu, S. Rawls, S. Harel, T. Hassner, I. Masi, J. Choi, J. Lekust, J. Kim, P. Natarajan, et al., Face recognition using deep multi-pose representations, in: Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on, IEEE, 2016, pp. 1–9.

[18] T. Hassner, S. Harel, E. Paz, R. Enbar, Effective face frontalization in unconstrained images, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 4295–4304.

[19] Y. Taigman, M. Yang, M. Ranzato, L. Wolf, Deepface: Closing the gap to human-level performance in face verification, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2014) 1701–1708.

[20] G. Hu, F. Yan, C.-H. Chan, W. Deng, W. Christmas, J. Kittler, N. M. Robertson, Face recognition using a unified 3d morphable model, in: European Conference on Computer Vision, Springer, 2016, pp. 73–89.

[21] G. Hu, F. Yan, J. Kittler, W. Christmas, C. H. Chan, Z. Feng, P. Huber, Efficient 3d morphable face model fitting, Pattern Recognition 67 (2017) 366–379.

[22] S. Banerjee, J. Brogan, J. Krizaj, A. Bharati, B. R. Webster, V. Struc, P. J. Flynn, W. J. Scheirer, To frontalize or not to frontalize: Do we really need elaborate pre-processing to improve face recognition?, in: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2018, pp. 20–29.

[23] A. Tuan Tran, T. Hassner, I. Masi, G. Medioni, Regressing robust and discriminative 3d morphable models with a very deep neural network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5163–5172.

[24] D. Jiang, Y. Hu, S. Yan, L. Zhang, H. Zhang, W. Gao, Efficient 3d reconstruction for face recognition, Pattern Recognition 38 (6) (2005) 787–798.

[25] A. Asthana, T. K. Marks, M. J. Jones, K. H. Tieu, M. Rohith, Fully automatic pose-invariant face recognition via 3d pose normalization, in: Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE, 2011, pp. 937–944.

[26] W. Deng, J. Hu, Z. Wu, J. Guo, Lighting-aware face frontalization for unconstrained face recognition, Pattern Recognition 68 (2017) 260–271.

[27] M. Kan, S. Shan, H. Chang, X. Chen, Stacked progressive auto-encoders (spae) for face recognition across poses, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2014) 1883–1890.

[28] J. Yang, S. Reed, M. H. Yang, H. Lee, Weakly-supervised disentangling with recurrent transformations for 3d view synthesis, in: International Conference on Neural Information Processing Systems, 2015, pp. 1099–1107.

[29] L. Tran, X. Yin, X. Liu, Disentangled representation learning gan for pose-invariant face recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Vol. 4, 2017, p. 7.

[30] X. Yin, X. Yu, K. Sohn, X. Liu, M. Chandraker, Towards large-pose face frontalization in the wild, arXiv:1704.06244.

[31] R. Huang, S. Zhang, T. Li, R. He, Beyond face rotation: Global and local perception gan for photorealistic and identity preserving frontal view synthesis, arXiv:1704.04086.

[32] V. Blanz, T. Vetter, A morphable model for the synthesis of 3d faces, in: Conference on Computer Graphics and Interactive Techniques, 1999, pp. 187–194.

[33] X. Chai, S. Shan, X. Chen, W. Gao, Locally linear regression for pose-invariant face recognition, IEEE Transactions on Image Processing 16 (7) (2007) 1716–1725.

[34] A. Li, S. Shan, W. Gao, Coupled bias–variance tradeoff for cross-pose face recognition, IEEE Transactions on Image Processing 21 (1) (2012) 305–315.

[35] A. Sharma, Generalized multiview analysis: A discriminative latent space, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 2160–2167.

[36] M. Kan, S. Shan, H. Chang, X. Chen, Stacked progressive auto-encoders (spae) for face recognition across poses, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1883–1890.

[37] Y. C. Shih, S. Paris, C. Barnes, W. T. Freeman, Style transfer for headshot portraits, Acm Transactions on Graphics 33 (4) (2014) 1–14.

[38] T. Zhou, S. Tulsiani, W. Sun, J. Malik, A. A. Efros, View synthesis by appearance flow, in: European Conference on Computer Vision, 2016, pp. 286–301.

[39] R. Yeh, Z. Liu, D. B. Goldman, A. Agarwala, Semantic facial expression editing using autoencoded flow, arXiv:1611.09961.

[40] C. Liu, J. Yuen, A. Torralba, Sift flow: dense correspondence across scenes and its applications, IEEE Transactions on Pattern Analysis & Machine Intelligence 33 (5) (2011) 978–94.

[41] C. Sagonas, Y. Panagakis, S. Zafeiriou, M. Pantic, Robust statistical face frontalization, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 3871–3879.

[42] B. K. P. Horn, B. G. Schunck, Determining optical flow, Artificial Intelligence 17 (13) (1981) 185–203.

[43] B. D. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, in: International Joint Conference on Artificial Intelligence, 1981, pp. 674–679.

[44] W. E. L. Grimson, Computational experiments with a feature based stereo algorithm, IEEE Transactions on Pattern Analysis & Machine Intelligence PAMI-7 (1) (2009) 17–34.

[45] S. Avidan, Ensemble tracking, IEEE Transactions on Pattern Analysis & Machine Intelligence 29 (2) (2007) 261–271.

[46] T. Brox, C. Bregler, J. Malik, Large displacement optical flow, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 41–48.

[47] D. G. Lowe, Object recognition from local scale-invariant features, in: IEEE International Conference on Computer Vision, 1999, p. 1150.

[48] R. Yu, S. Saito, H. Li, D. Ceylan, H. Li, Learning dense facial correspondences in unconstrained images, arXiv:1709.00536.

[49] M. Jaderberg, K. Simonyan, A. Zisserman, et al., Spatial transformer networks, in: Advances in Neural Information Processing Systems, 2015, pp. 2017–2025.

[50] Y. Ganin, D. Kononenko, D. Sungatullina, V. Lempitsky, Deepwarp: Photorealistic image resynthesis for gaze manipulation, in: European Conference on Computer Vision, Springer, 2016, pp. 311–326.

[51] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in Neural Information Processing Systems, 2014, pp. 2672–2680.

[52] E. L. Denton, S. Chintala, R. Fergus, et al., Deep generative image models using a laplacian pyramid of adversarial networks, in: Advances in Neural Information Processing Systems, 2015, pp. 1486–1494.

[53] A. Dosovitskiy, J. Tobias Springenberg, T. Brox, Learning to generate chairs with convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1538–1546.

[54] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al., Photo-realistic single image super-resolution using a generative adversarial network, arXiv:1609.04802.

[55] C. Li, M. Wand, Combining markov random fields and convolutional neural networks for image synthesis, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2479–2486.

[56] X. Yu, F. Porikli, Face hallucination with tiny unaligned images by transformative discriminative neural networks, in: AAAI Conference on Artificial Intelligence, 2017, pp. 4327–4333.

[57] K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks, in: European Conference on Computer Vision, Springer, 2016, pp. 630–645.

[58] M. D. Zeiler, G. W. Taylor, R. Fergus, Adaptive deconvolutional networks for mid and high level feature learning, in: IEEE International Conference on Computer Vision, IEEE, 2011, pp. 2018–2025.

[59] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1026–1034.

[60] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: International Conference on Machine Learning, 2015, pp. 448–456.

[61] D. Yi, Z. Lei, S. Liao, S. Z. Li, Learning face representation from scratch, arXiv:1411.7923.

[62] A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, Computer Science.

[63] A. L. Maas, A. Y. Hannun, A. Y. Ng, Rectifier nonlinearities improve neural network acoustic models, in: International Conference on Machine Learning, Vol. 30, 2013, p. 1.

[64] E. Saber, A. M. Tekalp, Frontal-view face detection and facial feature extraction using color, shape and symmetry based cost functions, Elsevier Science Inc., 1998.

[65] G. Passalis, P. Perakis, T. Theoharis, I. A. Kakadiaris, Using facial symmetry to handle pose variations in real-world 3d face recognition., IEEE Transactions on Pattern Analysis & Machine Intelligence 33 (10) (2011) 1938–1951.

[66] G. Hu, P. Mortazavian, J. Kittler, W. Christmas, A facial symmetry prior for improved illumination fitting of 3d morphable model, in: International Conference on Biometrics, 2013, pp. 1–6.

[67] L. A. Gatys, A. S. Ecker, M. Bethge, A neural algorithm of artistic style, arXiv:1508.06576.

[68] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, S. P. Smolley, Least squares generative adversarial networks, arXiv:1611.04076.

[69] R. Gross, I. Matthews, J. Cohn, T. Kanade, S. Baker, Multi-pie, Image & Vision Computing 28 (5) (2010) 807–813.

[70] G. B. Huang, M. Mattar, T. Berg, E. Learned-Miller, Labeled faces in the wild: A database forstudying face recognition in unconstrained environments, Month.

[71] B. F. Klare, B. Klein, E. Taborsky, A. Blanton, Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1931–1939.

[72] I. Masi, S. Rawls, G. G. Medioni, P. Natarajan, Pose-aware face recognition in the wild, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4838–4846.

[73] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, Computer Science.

[74] X. Zhu, Z. Lei, J. Yan, D. Yi, S. Z. Li, High-fidelity pose and expression normalization for face recognition in the wild, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 787–796.

[75] D. Wang, C. Otto, A. K. Jain, Face search at scale., IEEE Transactions on Pattern Analysis & Machine Intelligence PP (99) (2016) 1–1.

[76] J. C. Chen, V. M. Patel, R. Chellappa, Unconstrained face verification using deep cnn features, in: Applications of Computer Vision, 2016, pp. 1–9.

[77] H. Zhao, O. Gallo, I. Frosio, J. Kautz, Loss functions for image restoration with neural networks, IEEE Transactions on Computational Imaging 3 (2017) 47–57.