Domain-transferred Face Augmentation Network

Hao-Chiang Shao[1], Kang-Yu Liu[2], Chia-Wen Lin*[,2], and Jiwen Lu[3]

[1] Dept. of Statistics and Information Science, Fu Jen Catholic University, Taiwan
[2] Dept. of Electrical Engineering, National Tsing Hua University, Taiwan
[3] Dept. of Automation, Tsinghua University, China

Abstract. The performance of a convolutional neural network (CNN) based face recognition model largely relies on the richness of labelled training data. However, it is expensive to collect a training set with large variations of a face identity under different poses and illumination changes, so the diversity of within-class face images becomes a critical issue in practice. In this paper, we propose a 3D model-assisted domain-transferred face augmentation network (DotFAN) that can generate a series of variants of an input face based on the knowledge distilled from existing rich face datasets of other domains. Extending from StarGAN's architecture, DotFAN integrates with two additional subnetworks, i.e., face expert model (FEM) and face shape regressor (FSR), for latent facial code control. While FSR aims to extract face attributes, FEM is designed to capture a face identity. With their aid, DotFAN can separately learn facial feature codes and effectively generate face images of various facial attributes while keeping the identity of augmented faces unaltered. Experiments show that DotFAN is beneficial for augmenting small face datasets to improve their within-class diversity so that a better face recognition model can be learned from the augmented dataset.

1 Introduction

Face recognition is one of the most considerable research topics in the field of computer vision. Benefiting from meticulously-designed CNN architectures and loss functions [1–3], the performance of face recognition models have been significantly advanced. The performance of a CNN-based face recognition model largely relies on the richness of labeled training data. However, collecting a training set with large variations of a face identity under different poses and illumination changes is very expensive, making the diversity of within-class face images a critical issue in practice. This is a considerable problem in developing a surveillance system for small to medium sized real-world applications. In such cases, each identity usually has only a few face samples (we call it **Few-Face learning problem**), so what dominates the recognition accuracy is the data processing strategy, rather than the face recognition algorithm.

A face recognition model may fail, if the training set is too anemic to support the model. To avoid this circumstance, our idea is to distill the knowledge within a rich data domain and then transfer the distilled knowledge to enrich an incomprehensive set of training samples in a target domain via domain-transferred

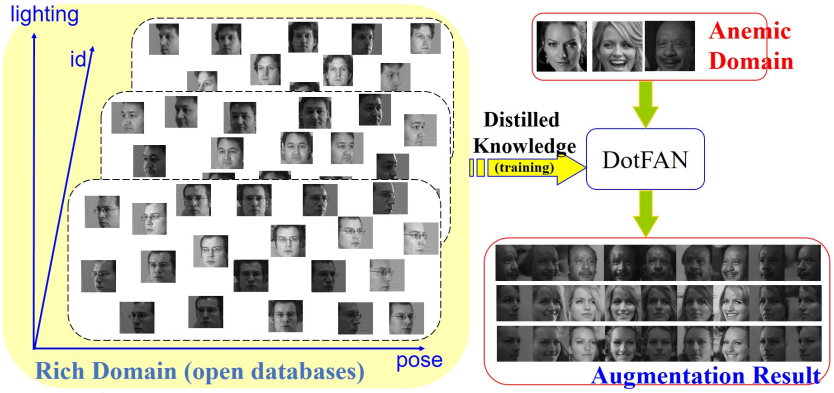

Fig. 1. DotFAN aims to enrich an anemic domain via identity-preserving face generation based on the knowledge, i.e., separated facial representation, distilled from data in a rich domain.

augmentation. Specifically, we aim to train a composite network, which learns a attribute-decomposed representation of faces from rich face datasets, so that this network can generate face variants—each associating with a different pose angle, a different facial expression, or a shading pattern due to different illumination condition—of each face subject in an anemic dataset for the data augmentation purpose. Hence, we propose in this paper a **Domain-transferred Face Augmentation Network (DotFAN)**, that aims to learn the distributions of the faces of distinct identities in the feature space from rich training data so that it can augment face data, including frontalized neutral faces, during inference by transferring the knowledge it learned, as its design concept illustrated in Fig. 1.

The proposed DotFAN is a face augmentation approach through which any identity class—no matter a minority class or not—can be enriched by synthesizing face samples based on the knowledge learned from rich face datasets of other domains via domain transfer. To this end, DotFAN first learns a facial representation from rich datasets to decompose the face information into essential facial attribute codes that are vital for identity identification and face manipulation. Then, exploiting this attribute-decomposed facial representation, DotFAN can generate synthetic face samples neighboring to the input faces in the sample space so that the diversity of each face-identify class can be significantly enhanced. As a result, the performance of a face recognition model trained on the enriched dataset can be improved as well.

Utilizing two auxiliary subnetworks, namely a data-driven face-expert model (FEM) [4, 5] and a model-assisted face shape regressor (FSR), DotFAN operates in a model-assisted data-driven fashion. FEM is a purely data-driven subnetwork pretrained on a domain rich in face identities, whereas FSR is driven by a 3D face model and pretrained on another domain with rich poses and expressions. Hence, FEM ensures that the synthesized variants of an input face are of the same identity as the input, while FSR collaborating with illumination code enables the model to generate faces with various poses, lighting (shading)

conditions, and different expressions. In addition, inspired by FaceID-GAN [6], we use a 3D face model (e.g., 3DMM [7]) to characterize face attributes related to pose and expression with only hundreds of parameters. Thereby, the size of FSR, and its training set of faces with labelled poses and expressions as well, is largely reduced, making it realizable with a light CNN with a much reduced number of parameters. Furthermore, the loss terms related to FEM and FSR act as regularizers during the training stage. This design prevents DotFAN from common issues in data-driven approaches, e.g. overfitting due to small training dataset.

Moreover, DotFAN is distinguishable from FaceID-GAN because of following reasons. First, based on a 3-player game strategy, FaceID-GAN regards its face-expert model as an additional discriminator that needs to be trained jointly with its generator and discriminator in an adversarial training manner. Because its face-expert model assists its discriminator rather than its generator, FaceID-GAN guarantees only the upper-bound of identity-dissimilarity. Also, this design may prevent FaceID-GAN's face expert model from pretraining and impede the whole training speeds. Furthermore, since it cannot be pretrained on a rich-domain data, this makes it difficult to transfer knowledge from a rich dataset to another in an on-line learning manner. On the contrary, DotFAN regards its FEM as a regularizer to guarantee that the identity information is not altered by the generator. Accordingly, FEM can be pretrained on a rich dataset and play a role of an inspector in charge of overseeing identity-preservability. This design not only carries out the identity-preserving face generation task, but also stabilizes and speeds up the training process by not intervening the competition between generator and discriminator. DotFAN has four primary contributions.

- We are the first to propose a domain-transferred face augmentation scheme that can easily transfer the knowledge distilled from a rich domain to an anemic domain, while preserving the identity of augmented faces in the target domain.
- DotFAN provides a learning-based universal solution for the **Few-Face** problem. Specifically, i) when a face recognizer is re-trainable, DotFAN enriches the **Few-Face Set** by data augmentation, and then the recognizer can be re-trained on the enriched set to improve its performance; and, ii) if the face recognizer is pretained on an incomprehensive dataset (e.g., with mainly frontal faces and/or neutral illumination) and is NOT re-trainable, DotFAN can assist the recognizer by frontalizing/neutralizing a to-be-recognized face.
- Through a concatenation of facial attribute codes learned separately from existing face datasets, DotFAN offers a unique unified framework that can incorporate prominent face attributes (pose, illumination, shape, expression) for face recognition and can be easily extended to other face related tasks.
- DotFAN well beats the state-of-the-arts by a significant gain margin in face recognition application with small-size training data available. This makes it a powerful tool for low-shot learning applications.

2 Related Work

Recently, various algorithms have been proposed to address the issue of small sample size with dramatic variations in facial attributes in face recognition [8–11]. This section reviews works on GAN-based image-to-image translation, face generation, and face frontalization/rotation techniques related to face augmentation.

(A) GAN-based image-to-image translation:

GAN and its variants have been widely adopted in a variety of fields, including image super-resolution, image synthesis, image style transfer, and domain adaptation. DCGAN [12] incorporates deep CNNs into GAN for unsupervised representation learning. DCGAN enables arithmetic operations in the feature space so that face synthesis can be controlled by manipulating attribute codes. The concept of generating images with a given condition has been adopted in succeeding works, such as Pix2pix [13] and CycleGAN [14]. Pix2pix requires pairwise training data to derive the translation relationship between two domains, whereas CycleGAN relaxes such limitation and exploits unpaired training inputs to achieve domain-to-domain translation. After CycleGAN, StarGAN [8] addresses the multi-domain image-to-image translation issue. With the aids of a multi-task learning setting and a design of domain classification loss, StarGAN's discriminator minimizes only the classification error associated to a known label. As a result, the domain classifier in the discriminator can guide the generator to learn the differences among multiple domains. Recently, an attribute-guided face generation method based on a conditional CycleGAN was proposed in [9]. This method synthesizes a high-resolution face based on an low-resolution reference face and an attribute code extracted from another high-resolution face. Consequently, by regarding faces of the same identity as one sub-domain of faces, we deem that face augmentation can be formulated as a multi-domain image-to-image translation problem that can be solved with the aid of attribute-guided face generation strategy.

(B) Face frontalization and rotation:

We regard the identity-preserving face rotation task as an inverse problem of the face frontalization technique used to synthesize a frontal face from a face image with arbitrary pose variation. Typical face frontalization and rotation methods synthesize a 2D face via 3D surface model manipulation, including pose angle control and facial expression control, such as FFGAN [15] and FaceID-GAN [6]. Still, some designs utilize specialized sub-networks or loss terms to reach the goal. For example, based on TPGAN [16], the pose invariant module (PIM) proposed in [17] contains an identity-preserving frontalization sub-network and a face recognition sub-network; the CNN proposed in [18] establishes a dense correspondence between paired non-frontal and frontal faces; and, the face normalization model (FNM) proposed in [5] involves a face-expert network, a pixel-wise loss, and a face attention discriminators to generate a faces with canonical-view and neutral expression. Finally, some methods approached this issue by means of disentangled representations, such as DR-GAN [19] and CAPG-GAN [20]. The former utilizes an encoder-decoder structure to learn a disentangled representa-

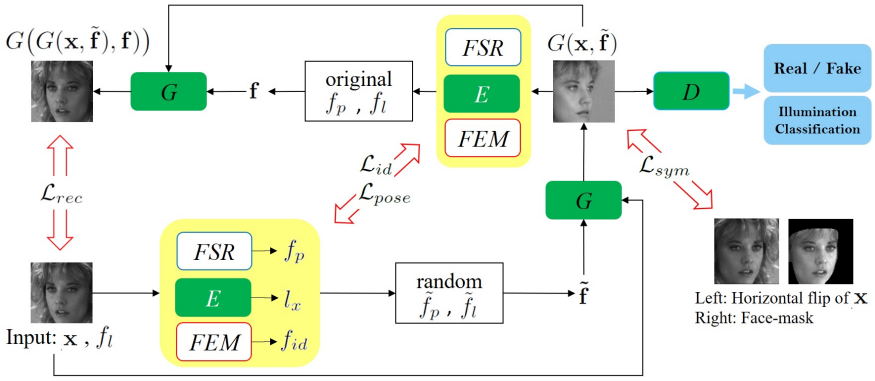

Fig. 2. Data flow of DotFAN's training process. FEM and FSR are independently pre-trained subnetworks, whereas E, G, and D are trained as a whole. $\tilde{f}_p$ and $\tilde{f}_l$ denote respectively a pose code and an illumination code randomly given in the training routine; and, f_l is the ground-truth illumination code provided by the training set. For inference, the data flow begins from $\mathbf{x}$ and ends at $G(\mathbf{x},\tilde{\mathbf{f}})$. Note that $\tilde{\mathbf{f}} = [l_x, f_{id}, \tilde{f}_p, \tilde{f}_l]$ and $\mathbf{f} = [E(G(\mathbf{x},\tilde{\mathbf{f}})), \Phi_{fem}(G(\mathbf{x},\tilde{\mathbf{f}})), f_p, f_l]$.

tion for face rotation, whereas the latter adopts a two-discriminator framework to learn simultaneously pose and identity information.

(C) Data augmentation for face recognition:
To facilitate face recognition, there are several face normalization and data augmentation methods. Face normalization methods aim to align face images by removing the volatility resulting from illumination variations, changes of facial expressions, and different pose angles [5], whereas the data augmentation method attempts to increase the richness of face images, often in aspects of pose angle and illumination conditions, for the training routine. To deal with illumination variations, conventional approaches utilized either physical models, e.g. Retinex theory [21], or 3D reconstruction strategy to remove/correct the shadow on a 2D image [22, 23]. Moreover, to mitigate the influence brought by pose angles, two categories of methods were proposed, namely pose-invariant face recognition methods and face rotation methods. While the former category focuses on learning pose-invariant features from a large-scale dataset [24, 25], the latter category, including face frontalization techniques, aims to learn the relationship between rotation angle and resulting face image via a generative model [15–17, 19, 20, 6]. Because face rotation methods are designed to increase the diversity of the view-points of face image data, they are also beneficial for augmentation tasks.

Based on these meticulous designs, DotFAN is implemented as an extension of StarGAN, involving an encoder-decoder framework and two sub-networks for learning attribute codes separately, and triggered by several loss terms, including reconstruction loss and domain classification loss, as will be elaborated later.

3 Domain-Transferred Face Augmentation

DotFAN is a framework to synthesize face images of one domain based on the knowledge, i.e., attribute-decomposed facial representation, learned from others. For a given input face $\mathbf{x}$, the generator G of DotFAN is trained to synthesize a face $G(\mathbf{x}, \mathbf{f})$ based on an input attribute code $\mathbf{f}$ comprising i) a general latent code $l_{\mathbf{x}} = E(x)$ extracted from $\mathbf{x}$ by the general facial encoder, ii) an identity code f_{id} indicating the face identity, iii) an attribute code f_p describing facial attributes including pose angle and facial expressions, and iv) an illumination code f_l. Through this design, a face image can be embedded via a concatenation of several attribute codes, i.e., $\mathbf{f} = [l_{\mathbf{x}}, f_{id}, f_p, f_l]$. Fig. 2 depicts the flow-diagram of DotFAN, and each component will be elaborated in following subsections.

3.1 Attribute-Decomposed Facial Representation

To obtain a decomposed representation, the attribute code $\mathbf{f}$ used by DotFAN for generating face variants is derived collaboratively by a general facial encoder E, a face-expert sub-network FEM, a shape-regression sub-network FSR, and an illumination code f_l. FEM and FSR are two well pre-trained sub-networks. FEM learns to extract identity-aware features from faces (of each identity) with various head poses and facial expressions, whereas FSR aims to learn pose features based on a 3D model. The illumination code is a 14×1 one-hot vector specifying 1 label-free case (corresponding to data from CASIA [26]) and 13 illumination conditions (associated with selected Multi-PIE dataset [27]).

(A) Face-Expert Model (FEM) Φ_{fem}: FEM Φ_{fem}, architecturally a ResNet-50, enables DotFAN to extract and to transplant the face identity from an input source to synthesized face images. Though conventionally face identity extraction is considered as a classification problem and optimized by using a cross-entropy loss, recent methods, e.g., CosFace [3] and ArcFace [2], proposed to adopt angular information instead. ArcFace maps face features onto a unit hyper-sphere and adjust between-class distances by using a pre-defined margin value so that a more discriminative feature representation can be obtained. Using ArcFace's loss function, FEM ensures not merely a fast training speed for learning face identity but also the efficiency in optimizing the whole DotFAN network.

(B) Face Shape Regressor (FSR): FSR, denoted as Φ_{fsr}, aims to extract face attributes including face shape, pose, and expression. Based on a widely used 3D Morphable Model (3DMM [7]), we designed our FSR as a model-assisted CNN rather than a fully data-driven network, which is complex and must be trained on a large variety of labeled face samples for characterizing face attributes because of the lack in prior knowledge. Because 3DMM can fairly and accurately characterize the face attributes using only hundreds of parameters, the model size of FSR can be significantly reduced. Firstly, we follow HPEN's strategy [28] to prepare ground-truth 3DMM parameters $\Theta_{\mathbf{x}}$ of an arbitrary face $\mathbf{x}$ from CASIA dataset [7]. Then, we train FSR via Weighted Parameter Distance Cost (WPDC) [29] defined in Eq. (1), with a modified importance matrix, as shown

in Eq. (2).

$$\mathcal{L}_{wpdc} = \big(\Phi_{fsr}(\mathbf{x}) - \Theta_{\mathbf{x}}\big)^t \mathbf{W} \big(\Phi_{fsr}(\mathbf{x}) - \Theta_{\mathbf{x}}\big) \tag{1}$$

$$\mathbf{W} = (w_R, w_T, w_{shape}, w_{exp}), \tag{2}$$

where w_R, $w_{t_{3d}}$, w_{shape}, and w_{exp} are distance-based weighting coefficients for the $\Theta_{\mathbf{x}}$ (consisting of a 9×1 vectorized rotation matrix R, a 3×1 translation vector T, a 199×1 vector α_{shape}, and a 29×1 α_{exp}) derived by 3DMM. Note that the facial attribute code $f_p = \Phi_{fsr}(\mathbf{x})$ extracted by FSR is a 240×1 vector mimicking $\Theta_{\mathbf{x}}$. While training DotFAN, we keep α_{shape}'s counterpart—representing facial shape—in f_p unchanged, and we replace f_p's other code segments corresponding to translation T, rotation R, and expression α_{exp} by arbitrary values.

(C) General facial encoder E and illumination code f_l:
E is used to capture other features, which cannot be represented by shape and identity codes, on a face. f_l is a one-hot vector specifying the lighting condition, based on which our model synthesizes a face. Note that because CASIA has no shadow labels, for f_l of a face from CASIA, its former 13 entries are set to be 0's and its 14^{th} entry $f_l^{casia} = 1$; this means to skip shading and to generate a face with the same illumination setting and the same shadow as the input.

3.2 Generator

The generator G takes an attribute code $\mathbf{f} = [l_{\mathbf{x}},\ f_{id},\ f_p,\ f_l]$ as its input to synthesize a face $G(\mathbf{x}, \mathbf{f})$. Described below are loss terms composing the loss function of our generator.

(A) Reconstruction loss:
In our design, we exploit a reconstruction loss to retain face contents after performing two transformations dual to each other. That is,

$$\mathcal{L}_{rec} = \|G\big(G(\mathbf{x}, \tilde{\mathbf{f}}), \mathbf{f}\big) - \mathbf{x}\|_2^2 / N, \tag{3}$$

where N is the number of pixels, $G(\mathbf{x}, \tilde{\mathbf{f}})$ is a synthetic face derived according to an input attribute code $\tilde{\mathbf{f}}$. This loss guarantees our generator can learn the transformation relationship between any two dual attribute codes.

(B) Pose-symmetric loss:
Based on a common assumption that a human face is symmetrical, a face with an x° pose angle and a face with a $-x^\circ$ angle should be symmetric about the 0° axis. Consequently, we design a pose-symmetric loss based on which DotFAN can learn to generate $\pm x^\circ$ faces from either training sample. This pose-symmetric loss is evaluated with the aid of a face-mask $M(\cdot)$, which is defined as a function of 3DMM parameters predicted by FSR and makes this loss term focus on the face region by filtering out the background, as described below:

$$\mathcal{L}_{sym} = \|M(\hat{\mathbf{f}}^-) \cdot \big(G(\mathbf{x}, \hat{\mathbf{f}}^-) - \hat{\mathbf{x}}^-\big)\|_2^2 / N. \tag{4}$$

Here, $\hat{\mathbf{f}}^- = [l_{\mathbf{x}}, f_{id}, \hat{f}_p^-, f_l]$, in which $\hat{f}_p^- = \Phi_{fsr}(\hat{\mathbf{x}}^-)$, and the other three attribute codes are extracted from $\mathbf{x}$. Additionally, $\hat{\mathbf{x}}^-$ is the horizontally-flipped

version of $\mathbf{x}$. In sum, this term measures the L_2-norm of the difference between a synthetic face and the horizontally-flipped version of $\mathbf{x}$ within a region-of-interest defined by a mask M.

(C) Identity-Preserving Loss:
We adopt the following identity-preserving loss to ensure that the identity code of a synthesized face $G(\mathbf{x}, \tilde{\mathbf{f}})$ is identical to that of input face $\mathbf{x}$. That is,

$$\mathcal{L}_{id} = \|\Phi_{fem}(\mathbf{x}) - \Phi_{fem}\big(G(\mathbf{x}, \tilde{\mathbf{f}})\big)\|_2^2 / N_1, \tag{5}$$

where N_1 denotes the length of $\Phi_{fem}(\mathbf{x})$.

(D) Pose-consistency loss:
This term guarantees that the pose and expression feature extracted from a synthetic face is consistent with $\tilde{f}_p$ used to generate the synthetic face. That is,

$$\mathcal{L}_{pose} = \|\tilde{f}_p - \Phi_{fsr}\big(G(\mathbf{x}, \tilde{\mathbf{f}})\big)\|_2^2 / N_2, \tag{6}$$

where N_2 denotes the length of $\tilde{f}_p$.

3.3 Discriminator

By regarding faces of the same identity as one sub-domain of faces, the task of augmenting faces of different identities becomes a multi-domain image-to-image translation problem addressed in StarGAN [8]. Hence, we exploit an adversarial loss to make augmented faces photo-realistic. To this end, we use the domain classification loss to verify if $G(\mathbf{x}, \tilde{\mathbf{f}})$ is properly classified to a target domain label f_l, which we used to specify the illumination condition of $G(\mathbf{x}, \tilde{\mathbf{f}})$. In addition, in order to stabilize the training process, we adopted the loss design used in WGAN-GP [30]. Consequently, these two loss terms can be expressed as follows:

$$\begin{aligned} \mathcal{L}_{adv}^{D} &= D_{src}(G(\mathbf{x}, \tilde{\mathbf{f}})) - D_{src}(\mathbf{x}) + \lambda_{gp} \cdot \big(\|\nabla_{\hat{x}} D_{src}(\hat{x})\|_2 - 1\big)^2 \\ \mathcal{L}_{adv}^{G} &= -D_{src}(G(\mathbf{x}, \tilde{\mathbf{f}})), \end{aligned} \tag{7}$$

where λ_{gp} is a trade-off factor for the gradient penalty, $\hat{x}$ is uniformly sampled from the linear interpolation between $\mathbf{x}$ and synthesized $G(\mathbf{x}, \tilde{\mathbf{f}})$, and D_{src} reflects a distribution over sources given by the discriminator; and,

$$\begin{aligned} \mathcal{L}_{cls}^{D} &= -\log D_{cls}(f_l|\mathbf{x}) \\ \mathcal{L}_{cls}^{G} &= -\log D_{cls}(\tilde{f}_l|G(\mathbf{x}, \tilde{\mathbf{f}})), \end{aligned} \tag{8}$$

where f_l is the ground-truth illumination code of $\mathbf{x}$, and $\tilde{f}_l$ is the illumination code embedded in $\tilde{\mathbf{f}}$.

In sum, the discriminator aims to produce probability distributions over both source and domain labels, i.e., $D : \mathbf{x} \rightarrow \{D_{src}(\mathbf{x}),\ D_{cls}(\mathbf{x})\}$. Empirically, $\lambda_{gp} = 10$.

3.4 Full objective function

In order to optimize the generator and alleviate the training difficulty, we pre-trained FSR and FEM with corresponding labels. Therefore, while training the generator and the discriminator, no additional label is needed. The full objective functions of DotFAN can be expressed as:

$$\begin{aligned}\mathcal{L}_G &= \mathcal{L}_{adv}^G + \mathcal{L}_{cls}^G + \mathcal{L}_{id} + \mathcal{L}_{pose} + \mathcal{L}_{sym} + \mathcal{L}_{rec} \\ \mathcal{L}_D &= \mathcal{L}_{adv}^D + \mathcal{L}_{cls}^D.\end{aligned} \tag{9}$$

Two loss terms in $\mathcal{L}_D$ are equal-weighted; and, the weighting factors of terms in $\mathcal{L}_G$ in turn are 1, 1, 8, 6, 5, and 5. Note that the alternative training of generator and discriminator was performed with ratio $1:1$.

4 Experimental Results

4.1 Dataset

DotFAN is trained jointly on **CMU Multi-PIE** [27] and **CASIA** [26]. Multi-PIE contains more than $750,000$ images of 337 identities, each with 20 different sorts of illumination and 15 different poses. We select images of pose angles ranging in between $\pm 45°$ and illumination codes from 0 to 12 to form our first training set, containing totally $84,000$ faces. From this training set, DotFAN learns the representative features for a wide range of pose angles, illumination conditions, and resulting shadows. Our second dataset is the whole CASIA set that contains $494,414$ images of $10,575$ identifies, each having about 50 images of different poses and expressions. Since CASIA contains a rich collection of face identities, it helps DotFAN learn features for representing identities.

To evaluate the performance of DotFAN on face synthesis, four additional datasets are used: **LFW** [31], **IJB-A** [32], **SurveilFace-1**, and **SurveilFace-2**. LFW has $13,233$ images of $5,749$ identities; IJB-A has $25,808$ images of 500 identities; SurveilFace-1 has $1,050$ images of 73 identities; and SurveilFace-2 contains $1,709$ images of 78 identities. We evaluate the performance of DotFAN's face frontalization on LFW and IJB-A. Besides, because faces in two SurveilFace datasets are taken in uncontrolled real working environments, they are contaminated by strong backlight, motion blurs, extreme shadow conditions, or influences from various viewpoints. Hence, they mimic the real-world conditions and thus are suitable for evaluating the face augmentation performance. The two SurveilFace sets are private data provided by a video surveillance provider. We will make them publicly available after removing personal labels.

We exploit CelebA to simulate the data augmentation process. CelebA contains $202,599$ images of $10,177$ identities with 40 kinds of diverse binary facial attributes. We randomly select a fixed number of images of each face identity from CelebA to form our simulation set, called "**sub-CelebA**" and conducted data augmentation experiments on both CelebA and sub-CelebA by using DotFAN.

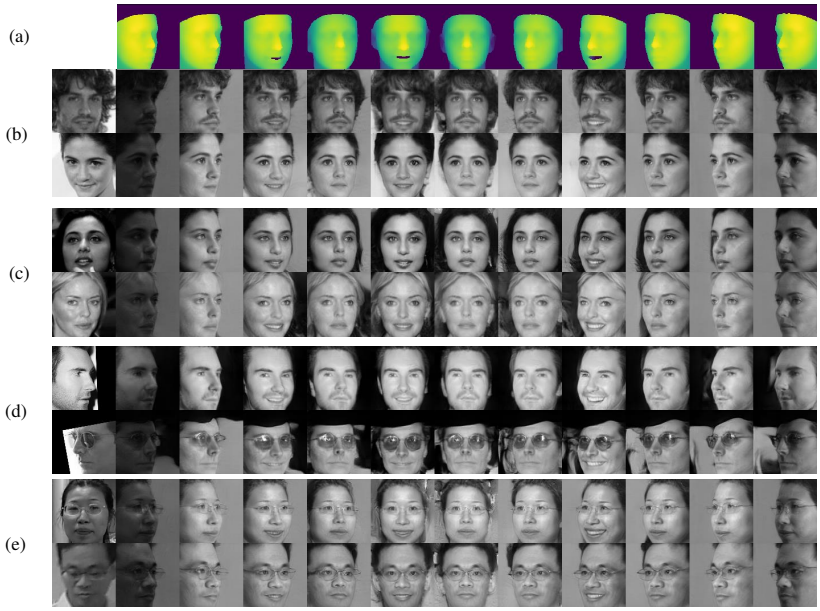

Fig. 3. Synthesized faces for face samples from different datasets generated by DotFAN. The left-most column shows the inputs with random attributes (e.g., poses, expressions, and motion blurs). The top-most row illustrates 3D templates with specific poses and expressions. To guarantee the identity information of each synthetic face is observable, columns 3–11 show shadow-free results, and columns 2 and 12 show faces with shadows. (a) 3D templates. (b) CelebA, (c) LFW, (d) CFP, and (e) SurveilFace.

Moreover, we demonstrate all face images in grayscale because of two reasons. First, two **SurveilFace** datasets are all grayscale. Second, DotFAN was trained partially on Multi-PIE in which images have reddish color-drift, so the same color-drift may occur on faces generated by DotFAN. Because such color-drift never degrades the recognition accuracy, we decided not to demonstrate color faces to avoid misunderstanding.

4.2 Implementation Details

Before training, we align the face images in the Multi-PIE and CASIA by MTCNN [33]. Structurally, our FEM is obtained by Resnet-50 pretained on MS-Celeb-1M [34], and FSR is implemented by a MobileNet [35] pretained on CASIA. To train DotFAN, each input face is resized to 112×112. Both generator and discriminator exploit Adam optimizer [36] with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The total number of training iterations is $420,000$ with a batch-size of 28, and the number of training epochs is 12. The learning rate is initially set to be 10^{-4} and begins to decay after the 6-th training epoch.

4.3 Face Synthesis

We verify the efficacy of DotFAN through the visual quality of i) face frontalization and ii) face rotation results.

(A) Face frontalization: First, we verify if the identity information extracted from a frontalized face, produced by DotFAN, is of the same class as the identity of a given source face. Following [6], we measure the performance by using a face recognition model trained on MS-Celeb-1M. Next, we conduct frontalization experiments on LFW. Table 1 shows the comparison of face verification results of frontalized faces. This experiment set validates that i) compared with other methods, DotFAN achieves comparable visual quality in face frontalization, ii) shadows can be effectively removed by DotFAN, and iii) both both DotFAN and casia-DotFAN (i.e., a DotFAN trained only on CASIA dataset) outperform other methods in terms of verification accuracy, especially in the experiment on IJB-A shown in Table 1(b), where DotFAN reports a much better TAR, i.e., 89.3% on FAR@0.001 and 93.7% on FAR@0.01, than existing approaches.

(B) Face Rotation Fig. 3 demonstrates DotFAN's capability in synthesizing faces of given attributes, including pose angles, facial expressions, and shadows, while retaining the associated identities. The source faces presented in the left-most column in Fig. 3 come from four datasets, i.e., CelebA, LFW, CFP [37], and SurveilFace. CelebA and LFW are two widely-adopted face datasets; CFP contains images with extreme pose angles, e.g., $\pm 90°$; and, SurveilFace contains faces of variant illumination conditions and faces affected by motion-blurs. This experiment shows that DotFAN can stably synthesize visually-pleasing face images based on 3DMM parameters describing 3D templates. Finally, Fig. 4 shows some synthesized faces with shadows assigned with four different illumination codes. Note that all synthesized faces presented in this paper are produced by the same DotFAN model without manually data-dependent modifications.

Table 1. Verification Table. (a) Verification accuracy on LFW. (b) True-Accept-Rate (TAR) of verifications on IJB-A. Note that while DotFAN has an FEM trained on MS-Celeb-1M in our design, the FEM of casia-DotFAN was trained on CASIA dataset.

(a)

Method	Verification Accuracy
HPEN [28]	96.25±0.76
FF-GAN [15]	96.42±0.89
FaceID-GAN [6]	97.01±0.83
casia-DotFAN	**98.55±0.52**
DotFAN	**99.18±0.39**

(b)

Method	FAR@0.01	FAR@0.001
PAM [24]	73.3±1.8	55.2±3.2
DCNN [38]	78.7±4.3	-
DR-GAN [19]	77.4±2.7	53.9±4.3
FF-GAN [15]	85.2±1.0	66.3±3.3
FaceID-GAN [6]	87.6±1.1	69.2±2.7
casia-DotFAN	**90.5±0.7**	**82.3±2.4**
DotFAN	**93.7±0.5**	**89.3±1.0**

Fig. 4. Face augmentation examples (CelebA) containing augmented faces with 4 illumination conditions and 7 poses.

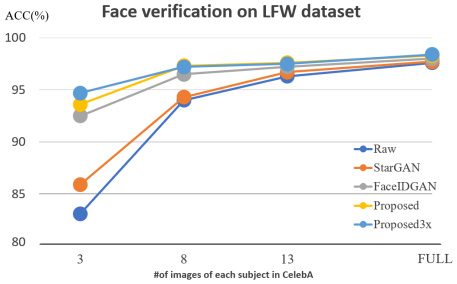

Fig. 5. Comparison of face verification accuracy on LFW trained on different augmented dataset. The horizontal spacing highlights the size of raw training dataset sampled from CelebA.

Fig. 6. Ablation study on loss terms. (a) Full loss. (b) w/o $\mathcal{L}_{id}$, (c) w/o $\mathcal{L}_{cls}$, (d) w/o $\mathcal{L}_{rec}$, (e) w/o $\mathcal{L}_{pose}$, and (f) w/o $\mathcal{L}_{sym}$.

4.4 Face Augmentation

Because DotFAN is a face augmentation network, experiments in this subsection were designed to show how face recognition accuracy can be improved with DotFAN-augmented training data. We adopted MobileFaceNet to be our face recognition network rather than other SOTAs because it's suitable to be deployed on mobile/embedded devices (less than 1M parameters) for small/medium sized real-world applications.

To evaluate the comprehensiveness of domain-transferred augmentation by DotFAN, we perform data augmentation on the same dataset by using DotFAN, FaceID-GAN, and StarGAN first; then, we compare the recognition accuracy of different MobileFaceNet models [39], each trained on an augmented dataset, by testing them on LFW and SurveilFace. StarGAN used in this experiment is trained on Mutli-PIE that is rich in illumination conditions; meanwhile, the FaceID-GAN is trained on CASIA to learn pose and expression representations.

Table 2 summarizes the results of this experiment set. We interpret the results focusing on Sub-experiment(a). In Sub-experiment(a), we randomly select 3 faces of each identity from CelebA to form the **RAW** training set, namely **Sub-CelebA(3)**, leading to about 30,000 training samples in raw Sub-CelebA(3). The MobileFaceNet trained on raw Sub-CelebA(3) achieves a verification accuracy of 83.1% on LFW, a true accept rate (TAR) of 20.5% at FAR = 0.001

Table 2. Performance comparison of face recognition models trained on different datasets. Here, **Sub-CelebA(x)** denotes a subset formed by randomly selecting x images of each face subject from CelebA

Method	LFW		SurveilFace-1			SurveilFace-2		
	ACC	AUC	@FAR=0.001	@FAR=0.01	AUC	@FAR=0.001	@FAR=0.01	AUC
(a) **Sub-CelebA(3)** (totally 30, 120 images)								
RAW	83.1	90.2	20.5	34.4	83.2	18.0	33.3	84.8
StarGAN	85.9	92.5	25.1	39.6	87.5	27.4	46.7	91.4
FaceID-GAN	92.5	97.6	34.6	53.5	92.8	32.3	54.0	94.3
Proposed 1x	93.6	98.1	35.7	56.2	93.6	34.7	57.8	95.0
Proposed 3x	94.7	98.7	36.8	58.3	94.6	36.5	60.8	95.6
(b) **Sub-CelebA(8)** (totally 75, 796 images)								
RAW	94.0	98.5	37.8	58.7	94.4	38.3	61.0	95.2
StarGAN	94.3	98.5	42.6	60.7	94.9	42.8	65.6	95.8
FaceID-GAN	96.5	99.3	48.1	65.6	96.0	45.7	67.9	96.8
Proposed 1x	97.3	99.5	53.2	71.2	97.0	49.1	72.2	97.2
Proposed 3x	97.2	99.5	53.2	68.9	96.9	47.3	70.0	97.1
(c) **Sub-CelebA(13)** (totally 116, 659 images)								
RAW	96.3	99.1	47.4	67.8	96.2	43.5	67.0	96.5
StarGAN	96.7	99.3	48.3	68.1	96.7	46.3	70.0	96.7
FaceID-GAN	97.2	99.5	53.3	71.3	97.0	50.2	72.3	97.4
Proposed 1x	97.6	99.6	56.2	75.1	97.7	50.4	73.9	97.7
Proposed 3x	97.5	99.7	56.7	75.5	97.7	53.9	72.2	97.8
(d) **CelebA (full CelebA dataset,** 202, 599 **images)**								
RAW	97.6	99.6	53.5	73.8	97.7	48.7	73.0	97.5
StarGAN	97.7	99.6	55.0	74.2	97.7	53.0	73.8	97.6
FaceID-GAN	98.0	99.7	57.6	76.4	98.1	54.1	76.5	98.0
Proposed 1x	98.3	99.8	62.4	80.9	98.4	57.1	76.7	98.1
Proposed 3x	98.4	99.7	61.4	78.9	98.2	54.7	77.8	98.0

on SurveilFace-1, and a TAR of 18.0% at FAR = 0.001 on SurveilFace-2. After giving each face in raw Sub-CelebA(3) a random facial attribute $\tilde{f}_p$ and a random illumination code $\tilde{f}_l$ to generate a new face and thus to double the size of the training set via DotFAN, the verification accuracy on LFW becomes 93.6%, and the TAR values on SurveilFace datasets are all nearly doubled, as shown in the row named **Proposed 1x**. This shows DotFAN is effective in face data augmentation and outperforms StarGAN and FaceID-GAN significantly. Furthermore, when we augment about 90, 000 additional faces to quadruple the size of training set, i.e., **Proposed 3x**, we have only a minor improvement in verification accuracy compared to **Proposed 1x**. This fact reflects that the marginal benefit a model can extract from the data diminishes as the number of samples increases when there is information overlap among data, as is what reported in [40]. Consequently, Table 2 and Fig. 5 reveal following remarkable points.

• First, by integrating attribute controls on pose angle, illuminating condition, and facial expression with an identity-preserving design, DotFAN outperforms StarGAN and FaceID-GAN in domain-transferred face augmentation tasks.

• Second, DotFAN's results obey *the law of diminishing marginal utility* in Economics[4] [41], as demonstrated in all **(Proposed 1x, Proposed 3x)** data pairs. Take LFW-experiment in Table 2(a) for example. An additional one-unit consumption of training data (1x-augmentation) brings an accuracy improvement, i.e., marginal utility, of 93.6%-83.1%=10.5%; when two more additional units

[4] This law primarily says that the marginal utility of each homogeneous unit decreases as the supply of units increases, and vice versa.

(3x-augmentation) are given, the improvement of accuracy is only 94.7%-93.6%= 1.1%. Therefore, a **1x** procedure is adequate to enrich a small dataset, and our experiments also show that the **Proposed 3x** procedure seems to reach the upper-bound of data richness.

• Third, although the improvement in verification accuracy decreases as the size of raw training set increases, DotFAN achieves a significant performance gain on augmenting a small-size face training set, as demonstrated in all (RAW, Proposed 1x) data pairs.

4.5 Ablation Study

We then verify the effect brought by each loss term. Fig. 6 depicts the faces generated by using different combinations of loss terms. The top-most row shows faces generated with the full generator loss $\mathcal{L}_G$ in Eq. (9), whereas the remaining rows respectively show synthetic results derived without one certain loss term.

As shown in Fig. 6(b), without $\mathcal{L}_{id}$, DotFAN fails to preserve the identity information although other facial attributes can be successfully retained. By contrast, without $\mathcal{L}_{cls}$, DotFAN cannot control the illumination condition, and the resulting faces all share the same shade (see Fig. 6(c)). These two rows evidence that $\mathcal{L}_{cls}$ and $\mathcal{L}_{id}$ are indispensable in DotFAN design. Moreover, Fig. 6(d) shows some unrealistic faces, e.g., a rectangular-shaped ear in the frontalized face; accordingly, $\mathcal{L}_{rec}$ is important for photo-realistic synthesis. Finally, Fig. 6(e)–(f) show that $\mathcal{L}_{pose}$ and $\mathcal{L}_{sym}$ are complementary to each other. As long as either of them functions, DotFAN can generate faces of different face angles. However, because $\mathcal{L}_{sym}$ is designed to learn only the mapping relationship between $+x^\circ$ face and $-x^\circ$ face by ignoring background outside the face region, artifacts may occur in the background region if $\mathcal{L}_{sym}$ works solely (see Fig. 6(e)).

5 Conclusion

We proposed a Domain-transferred Face Augmentation network (DotFAN) for generating a series of variants of an input face image based on the knowledge of attribute-decomposed face representation distilled from huge datasets. DotFAN is designed in StarGAN's style with two extra subnetworks to learn separately the facial attribute codes and produce a normalized face so that it can effectively generate face images of various facial attributes while preserving identity of synthetic images. Moreover, we proposed a pose-symmetric loss through which DotFAN can synthesize a pair of pose-symmetric face images directly at once. Extensive experiments demonstrate the effectiveness of DotFAN in augmenting small-sized face datasets and improving their within-subject diversity. As a result, a better face recognition model can be learned from an enriched training set derived by DotFAN.

References

1. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2016) 770–778
2. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2019) 4690–4699
3. Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., Liu, W.: CosFace: Large margin cosine loss for deep face recognition. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2018) 5265–5274
4. Cole, F., Belanger, D., Krishnan, D., Sarna, A., Mosseri, I., Freeman, W.T.: Synthesizing normalized faces from facial identity features. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2017) 3703–3712
5. Qian, Y., Deng, W., Hu, J.: Unsupervised face normalization with extreme pose and expression in the wild. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2019) 9851–9858
6. Shen, Y., Luo, P., Yan, J., Wang, X., Tang, X.: FaceID-GAN: Learning a symmetry three-player GAN for identity-preserving face synthesis. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2018) 821–830
7. Blanz, V., Vetter, T., et al.: A morphable model for the synthesis of 3D faces. In: Proc. ACM SIGGRAPH. (1999)
8. Choi, Y., Choi, M., Kim, M., Ha, J.W., Kim, S., Choo, J.: Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2018) 8789–8797
9. Lu, Y., Tai, Y.W., Tang, C.K.: Attribute-guided face generation using conditional CycleGAN. In: Proc. European Conf. Comput. Vis. (2018) 282–297
10. Li, T., Qian, R., Dong, C., Liu, S., Yan, Q., Zhu, W., Lin, L.: Beautygan: Instance-level facial makeup transfer with deep generative adversarial network. In: Proc. ACM Multimedia. (2018) 645–653
11. Shen, W., Liu, R.: Learning residual images for face attribute manipulation. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2017) 4030–4038
12. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)
13. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2017) 1125–1134
14. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Pro. IEEE Int. Conf. Comput. Vis. (2017) 2223–2232
15. Yin, X., Yu, X., Sohn, K., Liu, X., Chandraker, M.: Towards large-pose face frontalization in the wild. In: Proc. IEEE Int. Conf. Comput. Vis. (2017) 3990–3999
16. Huang, R., Zhang, S., Li, T., He, R.: Beyond face rotation: Global and local perception GAN for photorealistic and identity preserving frontal view synthesis. In: Proc. IEEE Int. Conf. Comput. Vis. (2017) 2439–2448
17. Zhao, J., Cheng, Y., Xu, Y., Xiong, L., Li, J., Zhao, F., Jayashree, K., Pranata, S., Shen, S., Xing, J., et al.: Towards pose invariant face recognition in the wild. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2018) 2207–2216

18. Zhang, Z., Chen, X., Wang, B., Hu, G., Zuo, W., Hancock, E.R.: Face frontalization using an appearance-flow-based convolutional neural network. IEEE Trans. Image Process. **28** (2018) 2187–2199
19. Tran, L., Yin, X., Liu, X.: Disentangled representation learning GAN for pose-invariant face recognition. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2017) 1415–1424
20. Hu, Y., Wu, X., Yu, B., He, R., Sun, Z.: Pose-guided photorealistic face rotation. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2018) 8398–8406
21. Land, E.H., McCann, J.J.: Lightness and retinex theory. Josa **61** (1971) 1–11
22. Finlayson, G.D., Hordley, S.D., Drew, M.S.: Removing shadows from images. In: Proc. European Conf. Comput. Vis. (2002) 823–836
23. Wang, Y., Zhang, L., Liu, Z., Hua, G., Wen, Z., Zhang, Z., Samaras, D.: Face relighting from a single image under arbitrary unknown lighting conditions. IEEE Trans. Pattern Anal. Mach. Intell. **31** (2008) 1968–1984
24. Masi, I., Rawls, S., Medioni, G., Natarajan, P.: Pose-aware face recognition in the wild. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2016) 4838–4846
25. Cao, K., Rong, Y., Li, C., Tang, X., Loy, C.C.: Pose-robust face recognition via deep residual equivariant mapping. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2018) 5187–5196
26. Yi, D., Lei, Z., Liao, S., Li, S.Z.: Learning face representation from scratch. arXiv preprint arXiv:1411.7923 (2014)
27. Gross, R.and Matthews, I., Cohn, J., Kanade, T., Baker, S.: Multi-pie. Image Vis. Comput. **28** (2010) 807–813
28. Zhu, X., Lei, Z., Yan, J., Yi, D., Li, S.Z.: High-fidelity pose and expression normalization for face recognition in the wild. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2015) 787–796
29. Zhu, X., Lei, Z., Liu, X., Shi, H., Li, S.Z.: Face alignment across large poses: A 3D solution. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2016) 146–155
30. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of Wasserstein GANs. In: Proc. Adv. Neural Inf. Proc. Syst. (2017) 5767–5777
31. Huang, G.B., Mattar, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. (2008)
32. Klare, B.F., Klein, B., Taborsky, E., Blanton, A., Cheney, J., Allen, K., Grother, P., Mah, A., Jain, A.K.: Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2015) 1931–1939
33. Dai, J., He, K., Sun, J.: Instance-aware semantic segmentation via multi-task network cascades. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2016) 3150–3158
34. Guo, Y., Zhang, L., Hu, Y., He, X., Gao, J.: MS-celeb-1m: A dataset and benchmark for large-scale face recognition. In: Proc. European Conf. Comput. Vis. (2016) 87–102
35. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017)
36. Kinga, D., Adam, J.B.: A method for stochastic optimization. In: Proc. Int. Conf. Learn. Represent. Volume 5. (2015)
37. Sengupta, S., Chen, J.C., Castillo, C., Patel, V.M., Chellappa, R., Jacobs, D.W.: Frontal to profile face verification in the wild. In: Proc. IEEE Winter Conf. Appl. Comput. Vis. (2016) 1–9

38. Chen, J.C., Patel, V.M., Chellappa, R.: Unconstrained face verification using deep cnn features. In: Proc. IEEE Winter Conf. Appl. Comput. Vis. (2016) 1–9
39. Chen, S., Liu, Y., Gao, X., Han, Z.: Mobilefacenets: Efficient CNNs for accurate real-time face verification on mobile devices. In: Proc. Chinese Conf. Biometric Recognit. (2018) 428–438
40. Cui, Y., Jia, M., Lin, T.Y., Song, Y., Belongie, S.: Class-balanced loss based on effective number of samples. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2019) 9268–9277
41. Mankiw, N.G.: Principles of economics. Cengage Learning (2020)