entropy

Article

Optimizing Few-Shot Learning Based on Variational Autoencoders

Ruoqi Wei and Ausif Mahmood *

Department of Computer Science & Engineering, University of Bridgeport, Bridgeport, CT 06604, USA; ruoqiwei@my.bridgeport.edu

* Correspondence: mahmood@bridgeport.edu; Tel.:+1-(203)-576-4737

Abstract: Despite the importance of few-shot learning, the lack of labeled training data in the real world makes it extremely challenging for existing machine learning methods because this limited dataset does not well represent the data variance. In this research, we suggest employing a generative approach using variational autoencoders (VAEs), which can be used specifically to optimize few-shot learning tasks by generating new samples with more intra-class variations on the Labeled Faces in the Wild (LFW) dataset. The purpose of our research is to increase the size of the training dataset using various methods to improve the accuracy and robustness of the few-shot face recognition. Specifically, we employ the VAE generator to increase the size of the training dataset, including the basic and the novel sets while utilizing transfer learning as the backend. Based on extensive experimental research, we analyze various data augmentation methods to observe how each method affects the accuracy of face recognition. The face generation method based on VAEs with perceptual loss can effectively improve the recognition accuracy rate to 96.47% using both the base and the novel sets.

Keywords: deep learning; variational autoencoders ; data representation learning; generative models; unsupervised learning; few shot learning; latent space; transfer learning

Citation: Wei, R.; Mahmood, A. Optimizing Few-Shot Learning Based on Variational Autoencoders. *Entropy* **2021**, *23*, 1390. https://doi.org/10.3390/e23111390

Received: 21 September 2021
Accepted: 21 October 2021
Published: 24 October 2021

1. Introduction

The explosion of big data has provided enough training samples in the real world to facilitate the development of deep learning performance [1–3]. Moreover, as a result of the development of high-performance computing devices such as graphics processing units (GPUs) and CPU clusters in recent years, the training of large-scale deep learning models has been greatly improved for big data feature learning. Compared to the early K80 (2014) GPU, which has 13 streaming microprocessors (SMs) with 2496 CUDA cores, NVIDIA's latest flagship GPU A100 (2020) has 108 SMs with 6912 CUDA cores, resulting in an approximately 10-fold faster novel performance for deep learning [4,5]. At present, deep learning models are often successful with millions of model parameters and a large amount of labeled big data available for training. Deep learning has also fueled significant progress in a variety of computer vision problems, such as object detection [6–9], motion tracking [10–13], action recognition [14,15], human pose estimation [16–19], and semantic segmentation [20–23]. Great success has also been achieved in face recognition with convolutional neural networks (CNN) [24–29]. However, many applications of this kind of deep learning in face recognition can only be realized on the premise of having a large amount of labeled data. Moreover, in real life, due to restrictions such as data security management and labor costs, it is impractical to obtain such a large amount of labeled data.

Humans, after learning only a few images of a target, can recognize and, sometimes, can even perceptually recognize the same target without learning the target image. Inspired by the ability of humans to learn quickly from a small number of samples, the field of artificial intelligence (AI) is currently actively researching few-shot learning, to solve the problems caused by limited datasets, by imitating the process of rapid recognition of the human brain to bring AI applications closer to the actual real-world scene.

The purpose of few-shot learning (FSL) is to learn the classifier of new classes; each class provides only a few training examples [30–32]. For instance, regarding practical applications of face recognition, such as surveillance and security, the face recognition system should be able to recognize people who have only seen it a few times; that is, the machine has the ability to see and understand things in the same manner as humans. Among the few existing sample-learning methods, data augmentation is an important approach. It is a weakly supervised series of techniques aimed at expanding datasets with additional data points. However, it is very challenging for existing machine learning methods because this limited dataset does not represent the data variance which describes the degree of spread in the dataset. Unbalanced data distribution or lack of data will cause over-parameterization and over-fitting problems, resulting in a significant decrease in the effectiveness of deep learning results. In face recognition, in particular, the variance in facial attributes, such as wearing glasses, which may detract from the recognition of the eye region and the overall facial appearance. Another example is wearing a beard, which makes it difficult to capture the boundary of the face and other features around the jawline, and can cause significant intra-class differences between faces of the same person, thus seriously affecting face recognition performance. To solve these problems, the construction of deep generative models has been attempted to refine the data and to convert the original data into features, thereby increasing the intra-class variations of the dataset. The most recent deep generative networks are VAEs [33–35] and generative adversarial networks (GANs) [36]. VAEs do not suffer from the problems encountered in GANs, which mainly relate to nonconvergence causing mode collapse and difficulty of evaluation [36–38]. In addition, a key benefit of VAEs is the ability to control the distribution of the latent representation vector z, which can combine VAEs with representation learning to further improve the downstream tasks [35,39]. VAEs can learn the smooth latent representations of the input data [40] and can thus generate new meaningful samples in an unsupervised manner.

In this research, we attempted to apply the VAE to the few-shot learning problem due to the scarcity of labeled training data. We employed the architecture proposed by [41] to train a model with a base set based on transfer learning and then build a feature extractor. Then, we undertook fine-tuning to learn the actual label of the target using a novel image dataset from the data augmentation. A face dataset is divided into a base set and a one-shot set. The base set implies that each person has only one picture. The one-shot set also means each person has only one picture. It is important to note that there is no overlap between the base set and the one-shot set. Using transfer learning as the backend, we implemented various types of data generation to increase the intra-class variations of the base set, thereby achieving higher recognition accuracy. Our face data augmentation for few-shot learning based on VAEs is fundamentally important for improving the performance of neural networks in the following aspects: (1) It is inexpensive to generate a huge number of synthetic data points with annotations in comparison to collecting and labeling real data. (2) Synthetic data can be accurate, so it is consistent with the ground-truth by nature. (3) If controllable generation method is adopted, faces with specific features and attributes can be obtained. (4) Face data augmentation has some special advantages, such as generating faces without self-occlusion [42] and a balanced dataset with more intra-class variations [43].

The main contributions of our work include:

- We employ a generative approach using variational autoencoders (VAEs) to optimize few-shot learning tasks by generating new samples with more intra-class variations.
- We analyze various data augmentation methods to observe how each method affects the accuracy of face recognition.
- We can generate synthetic data which is accurate and consistent with the ground-truth by nature, balance the dataset, and manipulate specific face attributes of the generated faces.
- Our framework significantly improves the face recognition accuracy rate in few-shot learning scenarios (up to 96.47% for the LFW dataset).

The structure of our paper is as follows. Section 2 describes background studies to our research. Section 3 explains the research plan in detail: (1) proposed architecture overview, (2) deep convolutional networks, (3) generation networks, (4) verification networks, and (5) identification networks. Section 4 outlines the experiments with implementation details and results. The summary, conclusion, and future work are provided in Section 5.

2. Related Work

2.1. Few-Shot Learning

Few-shot learning was proposed to solve the problem of learning new classes in classifiers, where each class provides only a small number of training samples [30–32]. As a result of the development of deep learning techniques, the existing FSL methods can in the following: metric learning [44–47], which learns metrics/similarity of few-shot samples through deep networks; meta-learning [48–50], which learns a meta-model in multiple FSL tasks, and then the meta-model can be used to predict the weight of the model in a new FSL task; transfer learning [51,52], which uses pretrained weights as initialization; and data augmentation [53,54], which is a form of weak supervision [55] and aims to expand the few-shot sample dataset with additional data points. It should be noted that there is no absolute distinction between the four categories. In this paper, our idea utilizes triplet-loss-based metric learning for few-shot face verification, utilizes data augmentation to improve the few-shot face recognition, and builds upon the transfer learning backend.

2.2. Data Augmentation

Data augmentation [53,54,56] is a technique used to increase the amount of available training data. The simplest method in data augmentation is the expansion of the dataset through basic digital image processing [57]. Digital image processing [58] includes photometric transformations [59] and geometric transformations [60]. Image processing is a traditional but powerful image augmentation method. However, these methods generate only repeated versions of the original data, and the dataset lacks intra-class variations. Therefore, its application is primarily to uniformly transform the entire image, rather than transforming specific attributes of the face.

Model-based face data augmentation is used to fit a face model to the input face, and then generate faces with different attributes by changing the parameters of the fixed model. Commonly used model-based face data augmentation techniques can be divided into 2D active appearance models (2D AAMs) [61] and 3D morphable models (3DMMs) [62]. Compared to image processing, the model-based method can be used to generate intra-class variations such as pose transformation and expressions. However, one of the biggest challenges is the difficulty in generating the teeth and mouth of the human face because these models can only generate the surface of the skin, and not the eyes, teeth, and mouth cavity [53]. Another shortcoming is that when the head posture changes, the lack of occlusion area causes artifacts [54]. Therefore, it is difficult to reconstruct a complete and accurate face model from a single 2D image through model-based face data augmentation. In addition, the computation cost of this method is also very high. In this research, our generative model-based transformation method does not only deal with pose transformation and expression transfer, but also with realistic facial attributes transformation at an affordable cost.

2.3. Generative Models

The principle of the deep generative model in the generation of new data is to use distribution estimation and sampling [38,63]. The traditional deep generative model is the Boltzmann series, that is, deep belief networks (DBNs) [64] and deep Boltzmann machines (DBMs) [65]. However, one of their main limitations is the high computational cost during the operation process [3]. The latest deep generative networks are VAEs [33–35] and generative adversarial networks (GANs) [36,66]. VAEs do not suffer problems encountered in GANs, which are mainly nonconvergence causing mode collapse and difficulty of evaluation [36–38]. In addition, a key benefit of VAEs is the ability to control the distribution of the latent representation vector z, which can combine VAEs with representation learning to further improve the downstream tasks [35,39]. VAEs can learn the smooth latent representations of the input data [40] and can thus generate new meaningful samples in an unsupervised manner. These properties have allowed VAEs to enjoy success, especially in computer vision; for example, static image generation [67,68], zero-shot learning [69–71], image super-resolution [72,73], network intrusion detection [74–76], and semantic image inpainting [77,78]. Table 1 lists the advantages and disadvantages of the proposed VAE data augmentation method along with those of other methods. However, compared with GANs, the samples generated by VAEs tend to be blurred and of lower quality. Recently, a breakthrough was made in VAEs by employing a perceptual loss function instead of reconstruction loss. The perceptual loss function is based on the high-level features extracted from pretrained deep CNNs to train feed-forward networks. It captures perceptual differences and spatial correlations between output and ground-truth images and solves the problem of blurry figures, thus resulting in high image quality [79,80]. Therefore, in our research, we employ VAEs using perceptual loss to generate networks.

Table 1. Pros and cons of different data augmentation methods.

Methods	Pros	Cons
Basic Image Processing	Capable of Geometric Transformation such as rotation and flipping	Generate duplicated version of the original data
	Capable of Photometric Transformation such as color jittering and contrast adjustment	Lacks intra-class variations
Model-Based Transformation	Capable of generating 2D and 3D pose transformation	Only represents the skin surface and does not include eyes, teeth, and mouth cavity
	Synthesizes faces with different expressions	The artifacts caused by the missing of occluded regions when the head pose is changed
		Computationally expensive
Generative Adversarial Networks (GANs)	Capable of generating intra-class variations	Hard to evaluate
	Capable of generating high-quality image	Non-convergence causing mode collapse
Variational Autoencoders (VAEs)	Capable of learning smooth latent representations of the input data	The generated samples tend to be blurry compared to GANs.
	Decent theoretical guarantee	Posterior collapse

2.4. Transfer Learning

The methods of transfer learning [81] can be divided into inductive transfer learning, transductive transfer learning, and unsupervised transfer learning. Furthermore, the inductive transfer learning method can be summarized into four situations based on the "what to transfer". Among these, instance-based transfer learning refers to reweighting samples in the source domain and correcting the marginal distribution difference between it and the target domain. This method works best when the conditional distributions in the two domains are the same. Feature-representation transfer is suitable for homogeneous and heterogeneous problems. This method works best when the source and the target domains have the same label space. Parameter-transfer approach transfers knowledge through the shared parameters of the source and the target domain learner models. As the pretrained model on the source domain has learned a well-defined structure, the pretrained model can be transferred to the target model if the two tasks are related. Because fine-tuning requires much less labeled data, this approach can potentially save time, reduce costs, and help improve robustness. The basic assumption of the relational knowledge-transfer problem [50] is that there are some common relations between the data in the source and the target domains. Therefore, the knowledge to be transferred is the common relationship between the source and the target domains. In this research, we focus on parameter-transfer approach-based transfer learning. This means that fine-tuning the CNN parameters from a pretrained model using a target training dataset is a particular form of transfer learning.

2.5. Facial Attribute Manipulation

Facial attribute analysis [82,83] includes facial attribute estimation (FAE), which is used to identify whether there are facial attributes in a given image, and facial attribute manipulation (FAM), which is used to synthesize or remove specific facial attributes [84]. In this research, we focus on FAM. There are two main methods to undertake FAM using generative models: model-based methods and extra-condition-based methods [84]. Model-based methods can only edit an attribute during a training process, but the disadvantage is its significant computation costs. Furthermore, there are two kinds of extra-condition-based methods: (1) attribute vectors as extra conditions, which, with an extra input vector, rely on simple linear interpolation; and (2) reference exemplars as extra conditions, which directly learn the image-to-image translation along with attributes, and is a popular approach for unsupervised disentanglement learning. However, disentangling is not easy to achieve, and to obtain better disentangling, the quality of reconstruction must be sacrificed [85]. Therefore, in this research, we focus on the first method that takes an attribute vector as the guidance to manipulate the desired attribute. Specifically, by changing a specific face attribute vector, the attributes of the face can be updated accordingly, referred to as deep feature interpolation (DFI) [86].

3. Research Plan

3.1. Proposed Architecture Overview

In this study, we analyzed the data augmentation method based on a variational autoencoder (VAE) in order to improve the accuracy and robustness of few-shot face recognition. We also increased the size of the training dataset in various ways to observe how data augmentation affects identification accuracy.

The complete deep face recognition system can be divided into three modules [87]: (1) a face detector to locate faces in images, (2) a facial landmark detector that can align faces with normalized coordinates, and (3) the face recognition module. We only focus on the face recognition module throughout the remainder of this paper.

Furthermore, face recognition can be divided into face verification (which answers the question, is this the same person?) and face identification (which addresses the question, who is this person?) [41]. Face verification calculates one-to-one similarity to determine whether two images belong to the same face, whereas face recognition calculates one-to-many similarities to determine the specific identity of the face. The face recognition module includes the following processes: (1) face processing, (2) deep feature extraction, and (3) face matching or face identification. In this research, we present the data augmentation method, which is facial attribute manipulation (FAM) using VAEs. Second, we adopt a pretrained architecture of Inception ResNet v1 [88] as a CNN for deep feature extraction for face data augmentation, face verification, and face identification tasks.

The proposed idea is as follows (as shown in Figure 1): a face dataset is divided into a base set and one-shot set. The base set means that each person has just one picture. The one-shot set also means that each person has only one picture. It should be noted that there is no overlap between the basic set and the one-shot set. Although the pictures in the base set and the one-shot set belong to the same person, the difference is that there may be changes in expression or posture between them, so they may not be correctly recognized. The face identification network is first pre-trained on the base set, and then fine-tuned through the augmented one-shot set. However, the face identification accuracy may not be good enough. Because the accuracy of individual recognition is proportional to the number of training images for each person, we can increase the accuracy of the one-shot set by adding augmented data. Thus, we used the proposed data augmentation method to increase the one-shot set and studied the change in the identification accuracy by this method. However, the problem with this method is that the performance of face identification may decrease with several augmented data points because the identity information is not sufficient. This is particularly the case for those images generated by VAEs. In order to solve this problem, we use a verification network to filter images that cannot be recognized; that is, if the augmented data can successfully pass the verification network, we pick out a subset from it. Finally, we can use these qualified augmented one-shot sets to fine-tune the face identification network with the original base set and the one-shot set. We hope to use this method and transfer learning as the backend to achieve higher accuracy of few-shot face recognition.

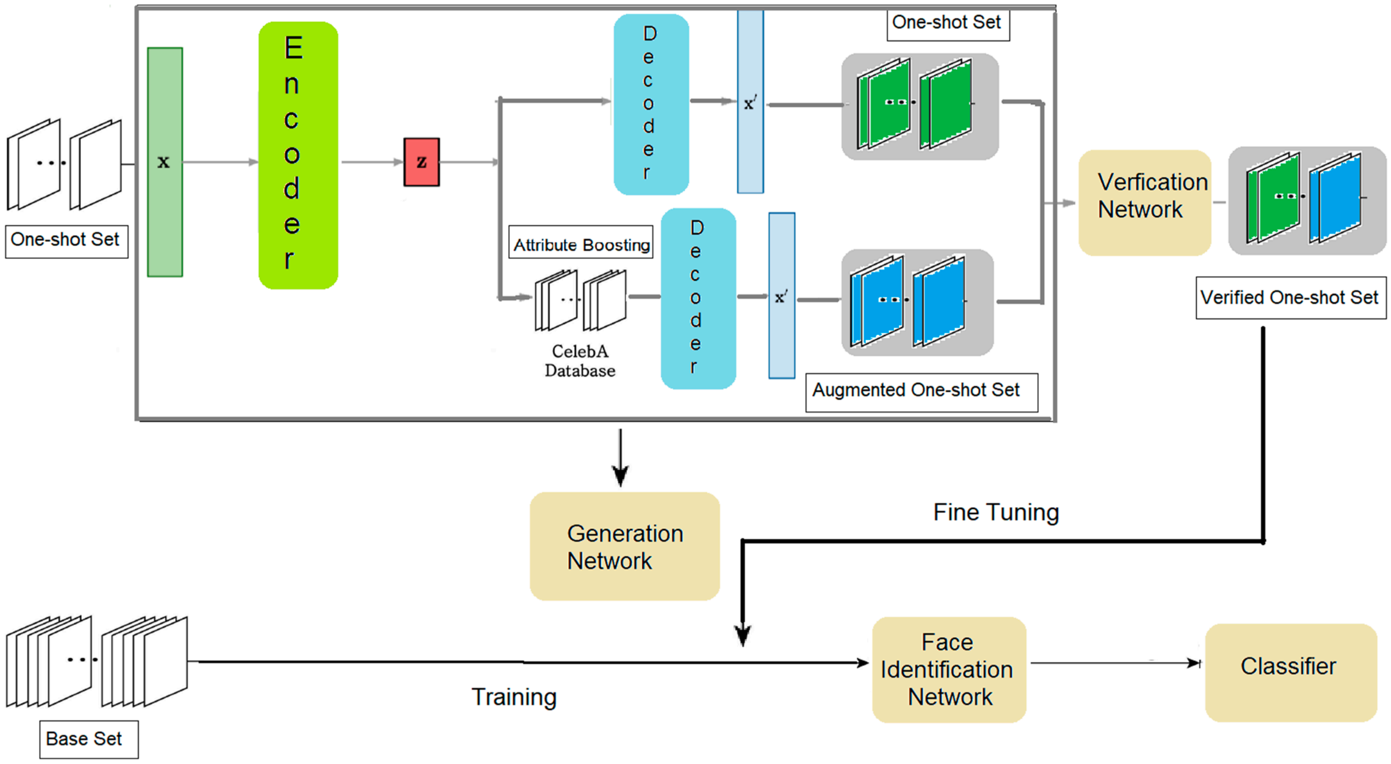

Figure 1. Proposed architecture overview.

3.2. *Deep Convolutional Networks*

We chose the best architecture for the results of the Computation Accuracy Tradeoff [41]—Inception Resnet V1—as the Deep Convolutional Network for the FaceNet system (as shown in Figure 2).

This deep neural network is almost the same as that described in [89]. The main difference between them is that the L_2 pooling is used in a local specific area instead of the maximum pooling (m). The pooling layer can reduce the number of parameters in the subsequent operation. The idea of L_2 pooling is to use L_2 regularized for pixel values in a local specific area, i.e., except for the final average pooling, the pooling is always 3×3 and is parallel to the convolution modules in each Inception module. If the dimensionality is reduced after pooling, it is represented by p. We then utilize 1×1, 3×3, and 5×5 pooling to concatenate and obtain the final output. Figure 2 describes the CNN network in detail. Note that all of our specific networks described in the next sections are based on this CNN framework.

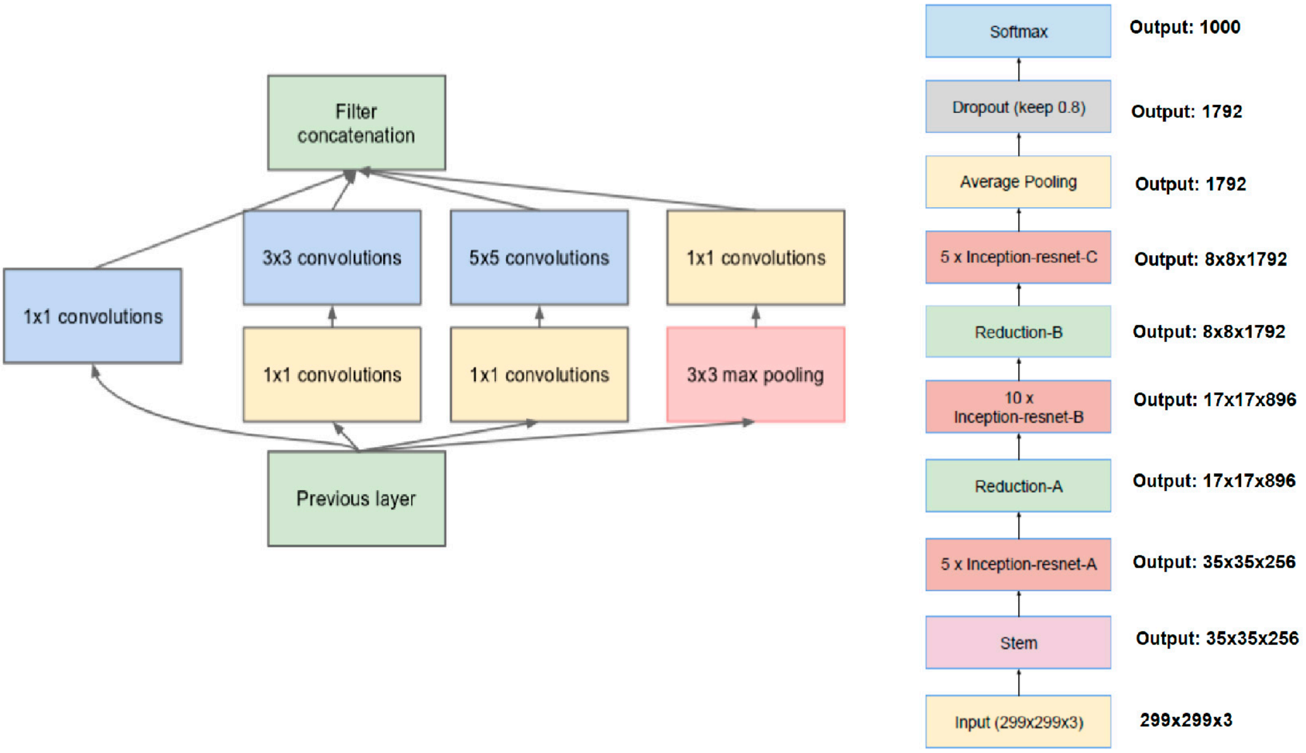

Figure 2. Inception module with dimension reductions (left) and schema for Inception-ResNet-v1 (right).

3.3. *Generation Network*

In our research, we employ VAE using FaceNet-based [41] perceptual loss similar to that in the paper [90] for face image generation with boosting attributes. Specifically, the pixel-by-pixel reconstruction loss of the deep convolutional VAE is replaced by a feature perceptual loss based on a pre-trained deep CNN. The feature perceptual loss is used to calculate the difference between the hidden representations of two images extracted from a pretrained deep CNN such as AlexNet [91] and VGGNet [92] trained on ImageNet [27]. This method attempts to improve the quality of the image generated by the VAE by ensuring the consistency of the hidden representation of the input image and the output image. It also imposes the spatial correlation consistency of the two images. The generative model consists of two parts—one is the autoencoder network, which includes an encoder network *E*(*x*) and a decoder network *D*(*z*), and the other is a pre-trained deep CNN, which is used to calculate the feature perceptual loss network ϕ. The encoder maps an input image x to a latent vector $z = E(x)$; then, the decoder maps the latent vector z back to image or data space $x' = D(z)$. After the VAE is trained, the decoder network can use the given vector z to generate a new image. We need two loss functions to train the VAE. First, after encoding an image x to a latent vector $z = Encoder(x) \sim q(z|x)$, the difference between the distribution of $q(z|x)$ and the distribution of a Gaussian distribution (called KL Divergence) can be minimized by the gradient descent algorithm: the first is KL divergence loss $\mathcal{L}_{KL} = D_{kl}[q(z|x)||p(z)]$ [33] which is used to ensure that the latent vector z is a Gaussian random variable. The other is feature perceptual loss, which computes the difference between hidden layer representations, i.e., $\mathcal{L}_{Rec} = \mathcal{L}_1 + \mathcal{L}_2 + \ldots + \mathcal{L}_n$, where $\mathcal{L}_n$ is the feature loss at the n^{th} hidden layer. During the training process, the pre-trained CNN network is fixed and is only used for advanced feature extraction. KL divergence loss $\mathcal{L}_{kl}$ is only used to update the encoder network, and feature perception loss $\mathcal{L}_{Rec}$ is used to update the encoder and decoder parameters.

3.3.1. Variational Autoencoder Network Architecture

The neural networks of the encoder and decoder are both constructed from deep CNN models such as AlexNet [91] and VGGNet [92]. As shown in Figure 3, this structure includes fully connected (FC) layers and convolutional (Conv) layers. The input image passes through four Conv layers and the last FC layer until the latent variable space is reached. The two convolutional layers in the encoder network achieve feature maps' dimensionality reduction using a stride of 2 and a 4 × 4 kernel size, which refers to the size of the filter that encodes a specific feature. After each Conv layer, there is a batch normalization layer and a LeakyReLU activation layer. Finally, two fully connected output layers (for μ and σ^2) are used to calculate KL divergence loss and sample latent variable z. The generative model $p(x|z)$ takes the sampled latent variables z received by μ and σ^2 and, using the reparameterization trick, feeds it through one FC layers and four Conv layers until a reconstructed output is obtained. Finally, it uses a stride of 1 and a kernel size of 3×3 in the deconvolution to obtain the reconstruction image. For upsampling, compared to the fractional-strided convolutions used in other works [23,63], we use the nearest neighbor method at a scale of 2. After each Conv layer, there is also a batch normalization layer and a LeakyReLU activation layer to help stabilize training.

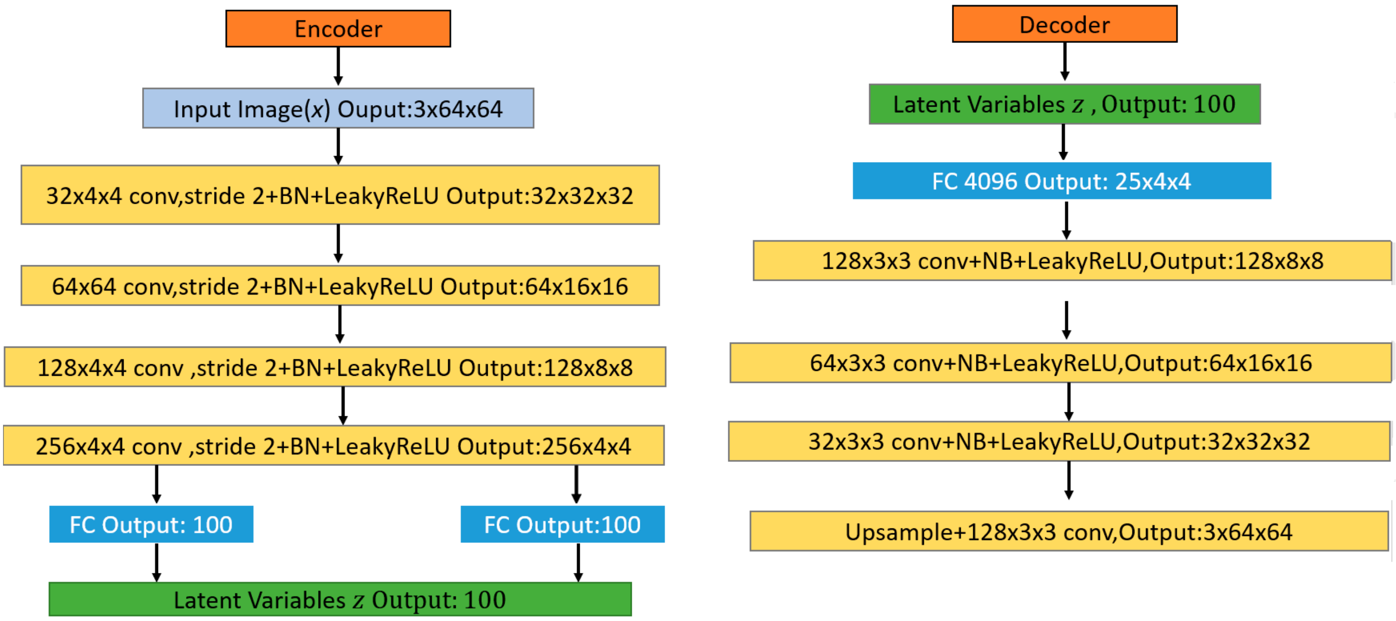
Encoder
Input Image(x) Ouput:3x64x64
32x4x4 conv,stride 2+BN+LeakyReLU Output:32x32x32
64x64 conv,stride 2+BN+LeakyReLU Output:64x16x16
128x4x4 conv ,stride 2+BN+LeakyReLU Output:128x8x8
256x4x4 conv ,stride 2+BN+LeakyReLU Output:256x4x4
FC Output: 100
FC Output:100
Latent Variables z Output: 100
Decoder
Latent Variables z , Output: 100
FC 4096 Output: 25x4x4
128x3x3 conv+NB+LeakyReLU,Output:128x8x8
64x3x3 conv+NB+LeakyReLU,Output:64x16x16
32x3x3 conv+NB+LeakyReLU,Output:32x32x32
Upsample+128x3x3 conv,Output:3x64x64

Figure 3. VAE architecture for the encoder network (left) and the decoder network (right).

3.3.2. Feature Perceptual Loss

The feature perception loss of two images is defined as the difference between hidden representations in the pre-trained deep CNN ϕ. We use Inception ResNet V1 as the Deep Convolutional Network for the FaceNet system in our experiment (as shown in Figure 4).

$$\mathcal{L}_{rec}^{n} = \frac{1}{2C^{n} \times W^{n} \times H^{n}} \sum_{c=1}^{C^{n}} \sum_{w=1}^{W^{n}} \sum_{h=1}^{H^{n}} (\phi(x)_{c,w,h}^{n} - \phi(x')_{c,w,h}^{n})^{2} \quad (1)$$

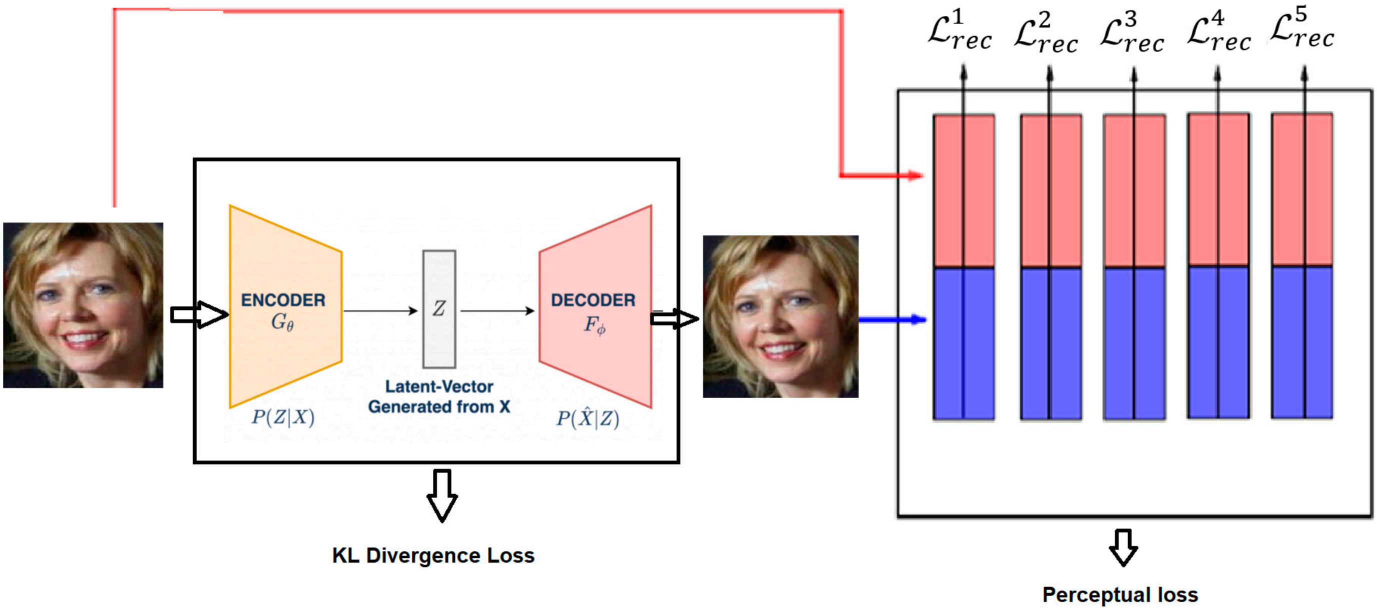
$\mathcal{L}_{rec}^{1}$ $\mathcal{L}_{rec}^{2}$ $\mathcal{L}_{rec}^{3}$ $\mathcal{L}_{rec}^{4}$ $\mathcal{L}_{rec}^{5}$
ENCODER
G_θ
Z
DECODER
F_ϕ
Latent-Vector Generated from X
$P(Z|X)$
$P(\hat{X}|Z)$
KL Divergence Loss
Perceptual loss

Figure 4. VAE with perceptual loss architecture overview.

The final reconstruction loss is the total loss obtained by adding the losses of the different layers of the deep CNN network, that is $\mathcal{L}_{rec} = \sum_n \mathcal{L}_{rec}^{n}$. In addition, we must add KL divergence loss $\mathcal{L}_{kl}$ to ensure that the latent vector z is a Gaussian random variable. Therefore, the training mode of this VAE model is to jointly minimize

The KL divergence loss $\mathcal{L}_{kl}$ and the total feature perceptual loss $\mathcal{L}_{rec}^{n}$ of the deep CNN network; that is,

$$\mathcal{L}_{total} = \alpha\mathcal{L}_{kl} + \beta \sum_{i}^{n} \mathcal{L}_{rec}^{n} \quad (2)$$

where α and β are weighting parameters for KL divergence and feature perceptual loss [93].

3.3.3. Attribute Boosting using VAEs

In order to be able to manipulate attributes such as face attributes and gender, the attribute vector needs to be calculated first, which can be achieved through simple vector arithmetic operations [94], thereby showing a rich linear structure in the representation space. A well-known example of vector arithmetic operations is vector('King')-vector('Man') + vector('Woman') resulting in a Queen's vector. In this study, we performed a similar arithmetic on the Z representation of our generators for visual concepts. In this paper, we investigate facial attributes of smiling. This is done by finding all images where the attribute 'Smiling' is not present and where the same attribute is present. The attribute vector is then calculated as the difference between the two average latent variables. Specifically, the images of two different attributes are sent to the encoder network to calculate the latent vectors, and the average latent vectors are calculated for each attribute, and are represented as $z_{pos\_smiling}$ and $z_{neg\_smiling}$. We can then use the difference $z_{pos\_smiling}$-$z_{neg\_smiling}$ as the latent vector $z_{smiling}$ of the smiling attribute. Finally, applying this smiling attribute latent vector to different latent vectors *z* to calculate a new latent vector *z*, such as $z + \alpha z_{smiling}$, where α= 0, 0.1 … 1, and then feeding the new latent vector *z* to the decoder network generates new face images.

3.4. Verification Network

After generating new faces with specific attributes, we can select the new faces if they can successfully pass the verification network. The verification network also uses the CNN architecture of FaceNet [41] (as shown in Figure 5). The loss function we use in the face verification model is the triplet loss; that is, we embed *f(x)* from the image *x* into the feature space R^d, and then minimize the squared distance between faces from the same identity and maximize the squared distance between faces from different identities [95].

The following sections describe: (1) learning face embedding with triplet loss; (2) triplet selection; and (3) the face verification task.

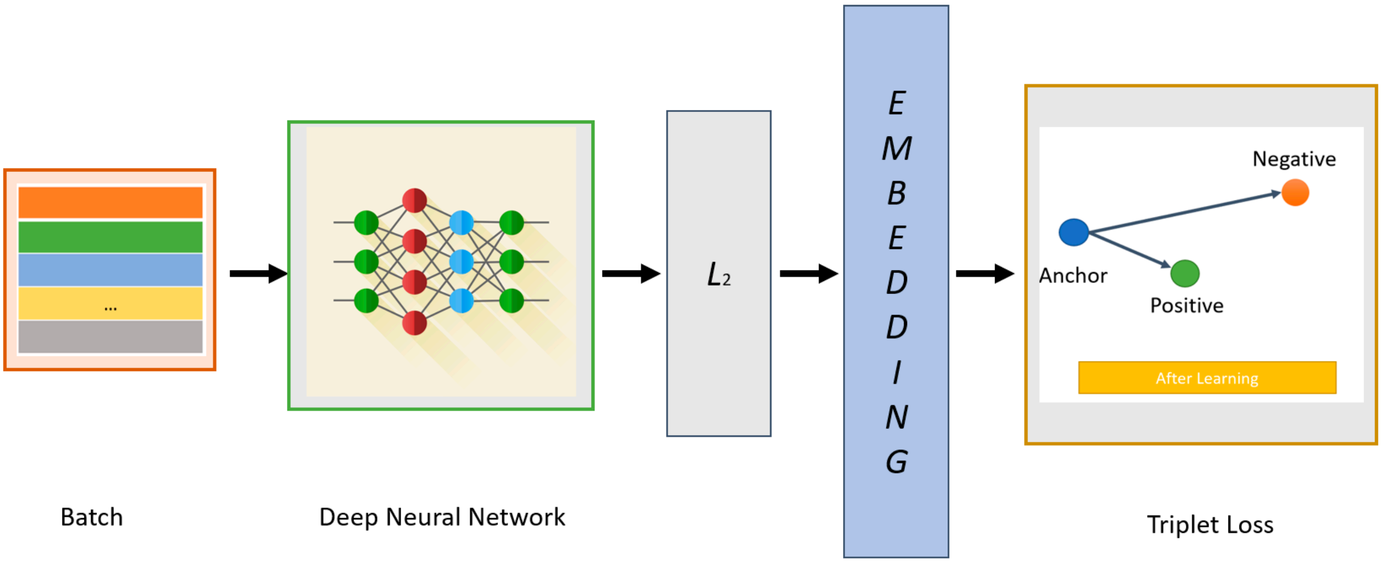

Figure 5. Face verification network architecture.

3.4.1. Learning Face Embedding with Triplet Loss

The idea of face embedding in this section is to embed the image *x* into a d-dimensional Euclidean space as $f(x) \in R^d$. It should be noted that this embedding is limited to the d-dimensional hypersphere, that is, the L_2 norm$||f(x)||_2 = 1$. This loss is caused by nearest neighbor classification [96]. Here, the network is trained to ensure that the output distance between an image x_i^a (anchor) and all other images x_i^p (positive) that belong to a known person is as close as possible, and that the distance between an image x_i^a (anchor) and any image x_i^n (negative) that belongs to an unknown person is as distant as possible (as shown in Figure 6). Thus,

$$||f(x_i^a) - f(x_i^p)||_2^2 + \alpha < ||f(x_i^a) - f(x_i^n)||_2^2, \ \forall((f(x_i^a), f(x_i^p), f(x_i^n)) \in T \quad (3)$$

where the threshold α is a margin that is enforced between positive and negative pairs. *T* is the set of all possible triplets. Then, the triplet loss is as follows:

$$\mathcal{L} = \sum_{i}^{N} [||f(x_i^a) - f(x_i^p)||_2^2 - ||f(x_i^a) - f(x_i^n)||_2^2 + \alpha] \quad (4)$$

3.4.2. Triplet Selection

There are two ways to achieve the selection of triplets, as follows:

1. Offline triplet mining: For example, at the beginning of each epoch, we calculate all embeddings on the training set, and then select only hard or semi-hard triples. Then, we can train the epoch with these triplets. However, this technique is not very effective because we need a complete pass of the training set to generate triplets. It also requires regular updates of triplets offline.
2. Online triplet mining: The idea here is to dynamically calculate useful triples for each batch of input. This technique allows you to provide more triples for a single batch of input without any offline mining and is more efficient. Therefore, we focus on online triplet mining. For example, if we want to generate triplets from these *B* embeddings, whenever we have three indexes *i*, *j*, *k* ∈ [1, *B*], if examples *i* and *j* have the same label but different images, and example k has different labels, then we say (*i*, *j*, *k*) is a valid triple.

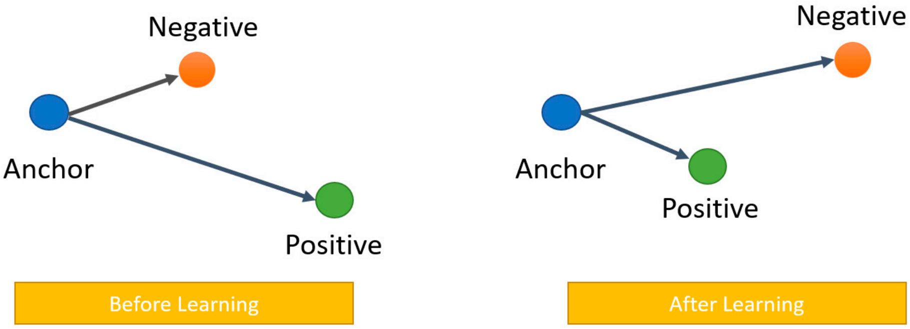

Figure 6. Triplet loss overview.

3.4.3. Face Verification Task

After learning the embedding using triplet loss with FaceNet's CNN, we can utilize this embedding for the FaceNet verification tasks. For example, the input is the paired faces with smiling and unsmiling faces, and the output is the L_2 distance through the verification network between the paired faces. If their L_2 distance is less than 1.1 [41], the pair of images are from the same person, otherwise they are not.

3.5. Identification Network

Finally, we can use these qualified augmented one-shot sets to fine-tune the face identification network with the original base set and the one-shot set. Similar to the verification network, our face identification network also combines the architecture and training strategy of the FaceNet with the deep convolutional networks and Softmax classifier.

The Softmax function takes a vector *z* of K real numbers as input and normalizes it to a probability distribution consisting of K probabilities that are proportional to the exponent of the input number. Before the *z* vector component is input to Softmax, the input *z* will not be in the interval (0,1). After Softmax is applied, however, each component will be in the interval (0,1), and the sum of the components is 1, so Softmax can convert the input *z* into a probability.

The Softmax function is defined by the formula:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_i}} \tag{5}$$

for $i = 1, \ldots, K$ and $z = (z_1,\ldots,z_K) \in \mathbb{R}^K$.

4. Experiments and Results

4.1. Datasets

We used the dataset of Labeled Faces in the Wild (LFW) [97] for training face identification network. The LFW dataset contains 13,233 face images with a total of 5749 identity labelled faces [98]. Among these, 1680 people have two or more images. The generative model VAE is trained by the CASIA-WebFace dataset [99], which has a total of 10,575 people and a total of 494,414 images. In the attribute boosting process, the CelebFaces Attributes Dataset (CelebA) [100] is utilized to extract attribute variables vectors. There are about 10k people in total, including 202,599 face images, and each image has 40 binary attribute annotations. Animals' faces (for example, dogs) can also be processed by the Deep Neural Networks. However, the size of the animals' face database will be enhanced and different conditions may be considered during the acquisition of an animal image for each subject: pose variation, distance variation, illumination variation, and occlusion (covering, non-covering) variation. Moreover, the currently available popular large-scale face recognition databases are all based on adults and there is no version for children; hence, our experiment is based on adults.

4.2. Implementation Details

4.2.1. Data Preprocessing and Face Alignment

Before the start of each training phase, each image data must be pre-processed to adapt to the face alignment network. We crop the rectangular image into a square with the side length of the short side of the original image. Then, we roughly align the image according to the eye position and adjust the image to 224 × 224-pixel RGB images.

The Multi-task Cascaded Convolutional Network (MTCNN) is a face detection and alignment method based on deep convolutional neural networks. This method can be used to complete face detection and alignment tasks at the same time. Pre-processing the images into 224×224-pixel RGB images is required before using the MTCNN. All face images are then detected, five key points are utilized to align the face images, and finally, all images are cropped and uniformly scaled to 160 × 160-RGB pixels.

4.2.2. Face Generation Using VAE

This section describes how to reconstruct face images with boosting attributes by the VAE using FaceNet-based perceptual loss.

(a) Train a VAE: This section describes how to train a VAE using perceptual loss. The VAE generation model is trained on the CASIA-WebFace Dataset. We use a batch size of 128 and 50,000 epochs to train the VAE model, and the NAdam method used to optimize [101] the initial learning rate is 0.01. The Inception ResNet CNN is used as the loss network ϕ to calculate the feature perception loss for image reconstruction. The size of the generated images is decided by the VAE implementation that generates 64 × 64-pixel images. The hardware specifications for executing implementations are a Tesla P100 GPU with 25 GB RAM. Table 2 shows some values of hyper-parameters used in this experiment.

(b) Calculate Attribute Vectors: In this step, the CelebA dataset is used to calculate vectors in latent variable space for several attributes. The CelebA dataset contains ~200k images annotated with 40 different attributes such as Blond Hair and Mustache. This is done by finding all images where the attribute is not present and where the same attribute is present. The attribute vector is then calculated as the difference between the two average latent variables. The VAE model checkpoint should point to the checkpoint trained in Step 1. Before running this, the CelebA dataset should also be aligned. The list_attr_celeba.txt file of the CelebA dataset contains the 40 attributes for each image and is available for download together with the dataset itself.

(c) Attribute Boosting Using VAEs: To demonstrate the usage of the VAE, after which we can modify attributes of an image, we apply the attribute-specific vector (calculated in step b) to different latent vectors z to calculate a new latent vector z, such as $z + \alpha z_{attribute}$, where $\alpha = 0, 0.1 \ldots 1$, and then feed the new latent vector z to the decoder network to generate new face images. Specifically, we select a few faces in the one-shot set where the specific attribute is not present. We then feed these images to the encoder of VAE, and then calculate the latent variables. We then add different amounts of the specific attribute vector (calculated in Step b) to different latent variables z and generate new faces with specific attribute images.

Table 2. Hyper-parameters used in all experiments.

HYPER-PARAMETERS	VALUES
EPOCHS	5000
BATCH-SIZE	128
LEARNING RATE	0.01
OPTIMIZER	Nadam
DATASET	LFW

4.2.3. Face Verification and Face Identification Experiments

The face verification function under the deepface interface offers to verify face pairs as the same person or different persons. This is a pre-trained network so there is no need to retrain. Table 2 shows some values of hyper-parameters to train the Softmax classifier for the Face Identification Network, which were used in all experiments. For comparison purposes, some parameters for all experiments were set to the same values to perform a fair comparison.

4.3. Results

4.3.1. Quality of The Reconstruction

In this research, we first used VAE based on feature perception loss to reconstruct face images. Figure 7 shows the result of the reconstruction. Top row: Input images. Bottom row: Generated images from VAE with feature perceptual loss. As shown in Figure 7, we can observe the difference between the face reconstructed by the VAE based on the feature perception loss and the original input face: the VAE based on the feature perception loss can not only generate human-like faces and the reconstructed face is similar to the original input face, but it can preserve the overall spatial face structure. We know that it is difficult for ordinary VAE to generate clear facial parts, such as eyes, nose, and mouth. This is because ordinary VAE tries to minimize the pixel-by-pixel loss between two images, whereas pixel-based loss does not contain perceptual and spatially related information. However, VAE based on the loss of feature perception can generate clear facial parts, such as eyes, nose, and mouth. This point of view is also confirmed in our experiments. Note that a too bright background will reduce the quality of the dark field on a face, thereby decreasing the quality of the reconstruction. When one captures images in low-light conditions, the images often suffer from low visibility. Moreover, images captured in low-light conditions usually suffer from very low contrast, which increases the difficulty of subsequent computer vision tasks to a great extent.

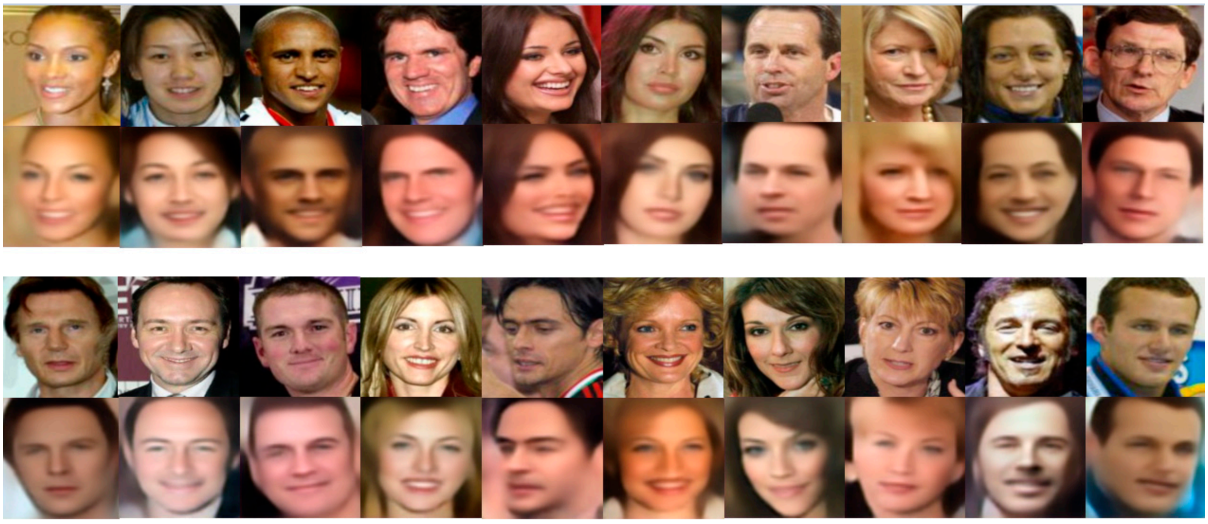

Figure 7. Face reconstruction by VAE with feature perception loss. Top row: Input images. Bottom row: Generated images from VAE with feature perceptual loss.

4.3.2. Pose Transition

In this experiment, we also tried to augment data by modifying images via rotating, flipping, adding noise, and jittering color. Our model includes these conventional schemes to provide prevention of overfitting that can occur within an inter-class, increase robustness, and achieve higher scores. The effectiveness of classical data augmentation is empirically shown in several previous studies. As shown in Figure 8, the first row is the original face, the second row is the augmented data by flipping, the third and the fourth rows are the results of adding noise and rotating, respectively, and the last row shows the augmented data by jittering color. In this way, each person has four different pose transitions; thus, in this step of the experiment, each person generates a total of four new pictures by basic image processing.

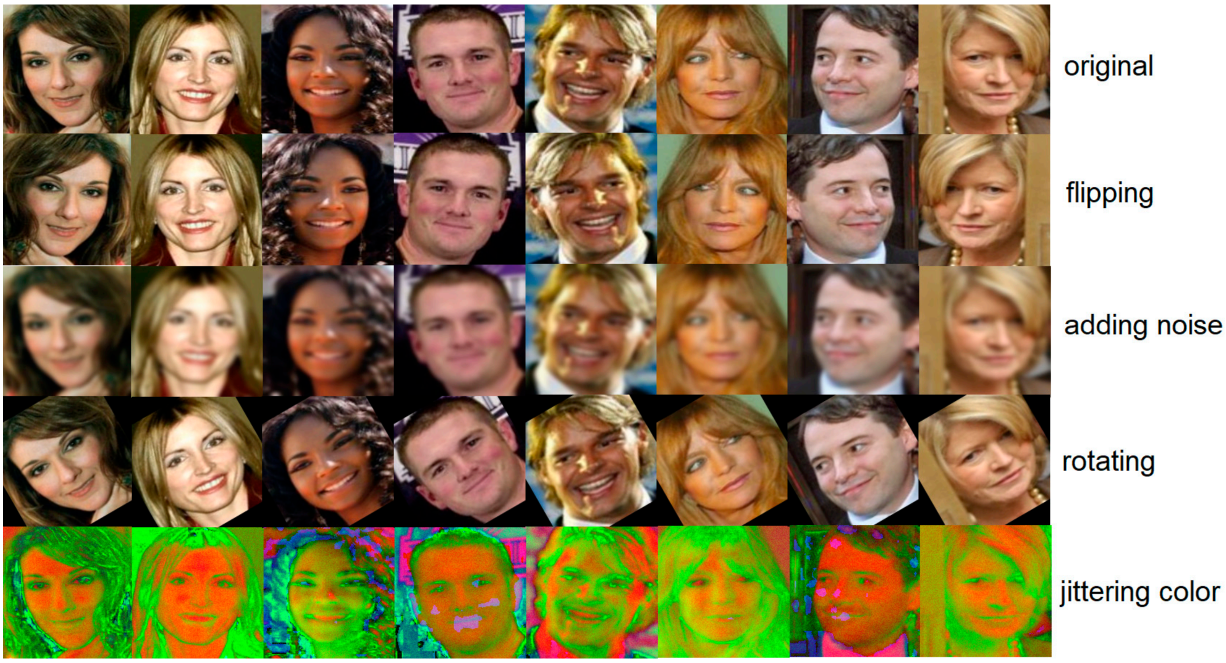

Figure 8. Face augmented data by modifying images via rotating, flipping, adding noise, and jittering color.

4.3.3. Attribute Manipulation

In this experiment, we did not seek to manipulate the overall face image, but aimed to control the specific attributes of the face image and generate a face image with specific attributes. By adding the vector of attributes to the face in the latent variable space, we can obtain the smooth transition process of adding this attribute to the face. As shown in Figure 9 (first row), by adding a smiling vector to the latent vector of non-smiling women, we can obtain a smooth transition from a non-smiling face to a smiling face. When the factor α increases, the appearance of the smile becomes more obvious, while other facial attributes can remain unchanged. Similarly, the second row is the transition process of adding a sunglasses vector from left to right, the third line and the fourth line are the results of adding a goatee vector and a heavy makeup vector, respectively, and the last line shows adding an attractive vector. In this manner, each person has 40 different attributes, and each attribute will generate 10 pictures. Thus, in this step of the experiment, each person will generate a total of 400 new pictures by VAE.

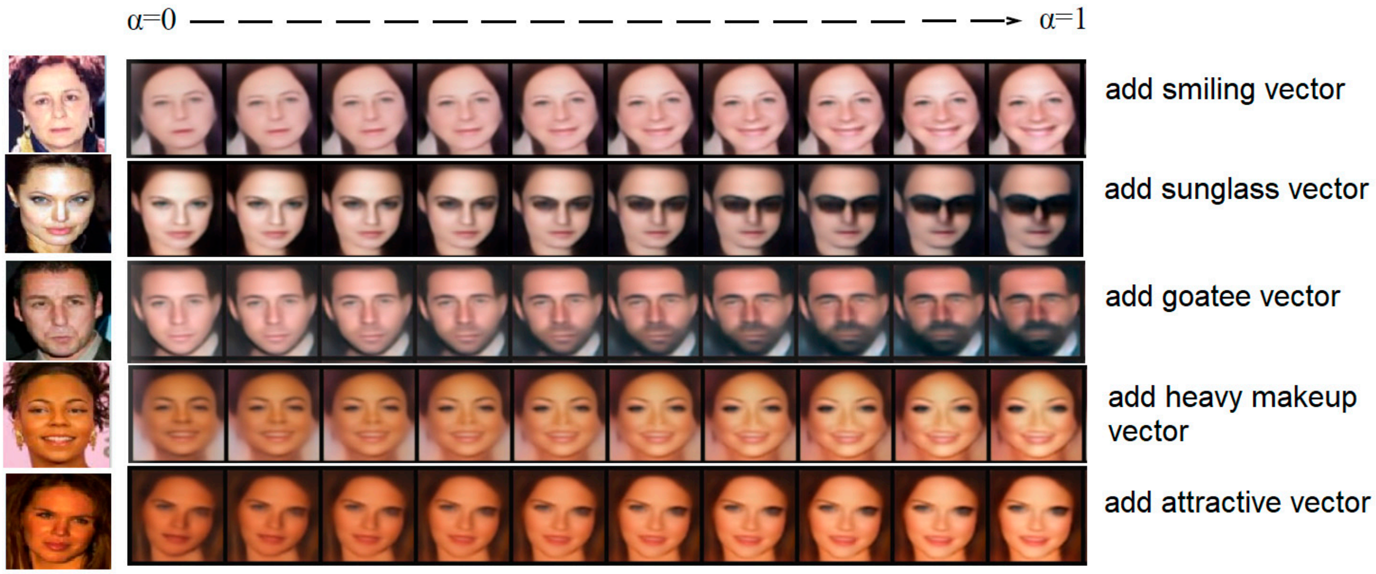

Figure 9. The vector arithmetic for visual attributes.

4.3.4. Correlation between Attribute-Specific Vectors

There are often correlations between different facial attributes. For example, heavy makeup and lipstick are often related to women. In order to study the correlation between different facial attributes in the CelebA dataset, we selected 15 facial attributes and calculated their attribute-specific latent vectors. Then, we used Pearson correlation to calculate the correlation matrix of these 15 attribute-specific vectors. Figure 10 shows the weight visualization of the learned correlation matrix. Red indicates a positive correlation, blue indicates a negative correlation, and the intensity of the color indicates the strength of the correlation. From the visualization results, we can find many related attribute pairs and many mutually exclusive attribute pairs. For example, the attributes of arched eyebrows and heavy makeup are given relatively high weights, indicating a positive correlation between these two attributes. The attributes of arched eyebrows and male are given relatively low weights, indicating a negative correlation between these two attributes. It is expected that women are generally considered to use more cosmetics than men. In addition, wearing a necklace seems to have no correlation with most other attributes, and there is only a weak positive correlation between wearing necklace, wearing lipstick, and wearing earring. This can also be explained because wearing a necklace, lipstick, and earrings are all facial decorations that often appear together.

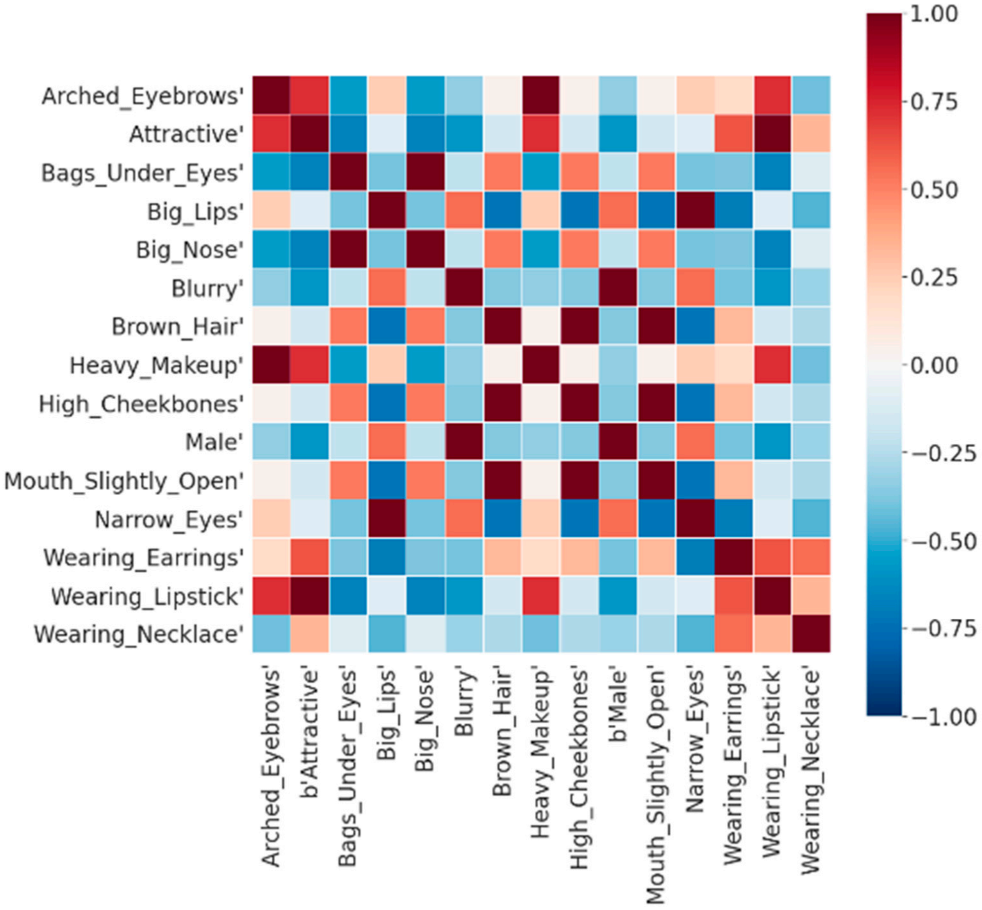

Figure 10. Pearson correlations between specific facial attributes.

4.3.5. Data Verification

Faces generated by deep generative models may occasionally fail to be recognized. For example, certain attributes (such as blond hair, young) added to the faces in the LFW dataset will result in a significant decrease in the face recognition rate. However, some attributes (such as pale skin, smiling) added to the faces in the LFW dataset can increase the face recognition rate. Therefore, to filter out unqualified augmented images and increase the face recognition rate, we used the verification network to verify the augmented data from the generation network, instead of manually selecting the appropriate attributes for the generation results. Table 3 shows the verification success rate of the generated images in the 1-shot experiment. At the same time, we also determined the verification success rate of the basic image processing data. It can be seen from the results that the verification rate of the basic image processing data is very high, further confirming that the use of basic image processing augmented data can enhance the robustness of our model and achieve higher scores. After this step, we can construct the final combined few-shot dataset: the novel image processing augmented dataset, the novel verified attribute augmented dataset, and the original basic dataset.

Table 3. Data verification results.

	Verification Rate (%)
Attribute boosting	10
Basic image processing	98.3

4.3.6. Data Identification

Our final face identification network was pre-trained by using the basic set, and then fine-tuned using the novel image processing augmented dataset and the novel verified attribute augmented dataset, so that individuals in the no overlap one-shot test set can be recognized. For evaluation, we measured identification accuracy. That is, suppose there are N images available in the test set, and C images are correctly recognized. Then, the accuracy is defined as C/N. We propose two methods of data augmentation and a combination of these two methods: (1) basic image processing; (2) attribute boosting using VAE; and (3) the combination of the above two methods. In these three methods, we gradually add new data to study the changes in identification accuracy.

The results of face identification are shown in Table 4. From these experiments, we observe that before data augmentation, the basic one-shot set was used to train the face identification network and the no overlap one-shot test set was used for testing. The final face identification accuracy rate is 88.52%. This shows that there is still significant scope for improvement in the accuracy of one-shot face identification. The best accuracy of one-shot face identification using VAE attribute boosting augmentation is 92.99%, indicating that although using VAE attribute boosting augmentation can improve the performance of one-shot face identification, it still lacks robustness and causes overfitting that can occur within an interclass. By comparison, the best accuracy of one-shot face identification using basic image processing is 93.67%. This proves that basic image processing methods can prevent possible over-fitting between classes. In many applications, they can be combined to improve the performance of the model [91,102]. Therefore, our model can utilize these basic image.

Processing methods to prevent overfitting that can occur within an interclass, increase robustness, and achieve higher scores. We observe that the combination of basic image processing and attribute boosting using VAE is more effective in improving performance than using classic basic image processing and VAE attribute boosting. It should be noted that the identification accuracy increased from 92.99% and 93.67% to 96.47%, respectively. This result shows that the use of VAE to produce more diverse training faces makes up for the insufficiency of the intra-class variation. Table 5 shows that when the L_2 distance of the verification network is set to less than 1.2 and 1.1 respectively, the results are sorted by L_2 distance from small to large. Then, we pick top1 to top5 faces and add them to the training set, and observe the results of the effect of the added faces on the identification accuracy of the no overlap one-shot set. From the results, we can observe that no matter whether the distance of L2 is set to less than 1.2 or 1.1, the identification accuracy is always higher than the result before the data augmentation, which is 88.52%. This proves that our verified VAE attribute augmented set improved the accuracy of identifying the one-shot set. In addition, although the identification accuracy is higher when the L2 distance is set to 1.2 compared rather than 1.1, after the image processing augmented set is added, the accuracy of the L2 distance set to 1.1 is higher than that set to 1.2. This shows that setting the L2 distance to 1.1 makes the combination of the verified VAE attribute augmented set and image processing augmented set more effective, thereby improving the robustness and generalization ability of the face identification model.

Table 4. Face identification results.

Training Method	**Test set**	**Identification Accuracy (%)**
basic one-shot set	no overlap one-shot set	88.52
basic one-shot set + proposed verified VAE attribute augmented set	no overlap one-shot set	92.99
basic one-shot set + image processing augmented set	no overlap one-shot set	93.67
basic one-shot set + image processing augmented set + proposed verified VAE attribute augmented set	no overlap one-shot set	96.47

Table 5. Face identification results with different L_2 distances.

Training Method	**Test set**	**Identification Accuracy (%)**				
		Top1-Shot	**Top2-Shot**	**Top3-Shot**	**Top4-Shot**	**Top5-Shot**
basic one-shot set + proposed verified VAE attribute augmented set with 1.1 L_2 distance	no overlap one-shot set	89.60	90.36	**90.69**	90.36	90.27
basic one-shot set + image processing augmented set+ proposed verified VAE attribute augmented set with 1.1 L_2 distance	no overlap one-shot set	95.70	95.52	95.67	**96.30**	95.36
basic one-shot set + proposed verified VAE attribute augmented set with 1.2 L_2 distance	no overlap one-shot set	90.66	**90.99**	90.83	90.40	90.87
basic one-shot set + image processing augmented set+ proposed verified VAE attribute augmented set with 1.2 L_2 distance	no overlap one-shot set	**95.83**	95.30	90.73	95.49	95.43

5. Conclusions

At present, few-shot learning technology based on image generation is highly attractive in various computer vision applications, particularly because of the lack of labeled data during training. In this research, we attempted to use VAE with feature perception loss to generate better visual quality face images with boosting attributes, and expand the training set to improve the performance of the few-shot face identification task. As the size of the dataset increased, we verified the change in the performance associated with the increased use of Inception Resnet V1 for few-shot face recognition. With these results, we can generate accurate synthetic data that is consistent with the ground-truth, balance the dataset with more intra-class variations, and manipulate specific face attributes of the generated faces. Moreover, it is inexpensive to generate such synthetic data with annotations in comparison to collecting and labeling real data. Although our research was rigorously executed and verified to achieve sufficient results, there are some limitations to synthetic data generation, as described below.

Firstly, although our proposed method improved the identification accuracy of few-shot learning from 88.56% to 96.47%, which is a significant change, there is still room for improvement. One method to decipher the behavior of deep neural network classifiers is by investigating their decision boundaries and their geometrical properties. Deep neural network classifiers are vulnerable to erroneous instances near their decision boundaries [103]. Therefore, in the future, we plan to determine decision boundaries using adversarial examples between the different classes and then generate instances near the decision boundary to further improve the few-shot face identification accuracy. In addition, although the VAE with feature perception loss contains perceptual and spatially related information and can generate clear facial parts, the generated samples tend to be of lower quality compared to those of GANs. The VAE-GAN [104] is a strategy built on the VAE structure with a GAN discriminator added after the decoder, which ensures that the samples generated by the VAE have high quality. Therefore, in the future, we plan to utilize the VAE-GAN with feature perception loss to further improve the quality of the reconstruction. Furthermore, in this research, we focused on the method that takes an attribute vector as the guidance for manipulating the desired attribute. The advantage of this method lies in the manipulation of multiple attributes by changing multiple corresponding condition values. However, this method cannot continuously change a certain attribute because the value of the attribute vectors is discrete. We believe that, in the future, this limitation can be resolved through interpolation schemes [105] or semantic component decomposition [106].

Finally, reference exemplar-based algorithms based on unsupervised disentanglement learning [107–109] are becoming a promising research direction. Compared with only manually changing the attribute vector, this method directly learns the image-to-image translation along with the attributes, and then manipulates these attributes using a simple traversal across regularization dimensions, so that images with more realistic details can be generated. For our future research, we plan to incorporate unsupervised disentanglement learning in our framework.

Author Contributions: This research is part of R.W.'s dissertation work under the supervision of A.M. All authors conceived and designed the experiments. R.W. performed the experiments. Formal analysis, R.W. Investigation, R.W. Methodology, R.W. Software, R.W. Supervision, R.W. Writing—original draft, R.W. Writing—review & editing, A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded in part by the UB Partners CT Next Innovation Grant 2019–2020.Also, this research work was funded in part by the Department of Computer Science and Engineering, University of Bridgeport, CT, USA

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: http://vis-www.cs.umass.edu/lfw/ (accessed on 23 October 2021.)

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning;* MIT press: Cambridge, MA, USA, 2016.
2. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444.
3. Voulodimos, A.; Doulamis, N.; Doulamis, A.; Protopapadakis, E. Deep learning for computer vision: A brief review. *Comput. Intell. Neurosci.* **2018**, *2018*, doi:10.1155/2018/7068349.
4. Choquette, J.; Gandhi, W.; Giroux, O.; Stam, N.; Krashinsky, R. Nvidia a100 tensor core gpu: Performance and innovation. *IEEE Micro* **2021**, *41*, 29–35.
5. Svedin, M.; Chien, S.W.; Chikafa, G.; Jansson, N.; Podobas, A. Benchmarking the Nvidia GPU Lineage: From Early K80 to Modern A100 with Asynchronous Memory Transfers. In Proceedings of the Proceedings of the 11th International Symposium on Highly Efficient Accelerators and Reconfigurable Technologies, Berlin, Germany,21 Jun,2021; pp. 1–6.
6. W. Wang, Q. Lai, H. Fu, J. Shen, H. Ling and R. Yang, "Salient Object Detection in the Deep Learning Era: An In-depth Survey," in IEEE Transactions on Pattern Analysis and Machine Intelligence, doi: 10.1109/TPAMI.2021.3051099.
7. Liu, Y.; Sun, P.; Wergeles, N.; Shang, Y. A survey and performance evaluation of deep learning methods for small object detection. *Expert Syst. Appl.* **2021**, *172*, 114602.
8. Ouyang, W.; Zeng, X.; Wang, X.; Qiu, S.; Luo, P.; Tian, Y.; Li, H.; Yang, S.; Wang, Z.; Li, H. DeepID-Net: Object detection with deformable part based convolutional neural networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1320–1334.
9. Diba, A.; Sharma, V.; Pazandeh, A.; Pirsiavash, H.; Van Gool, L. Weakly supervised cascaded convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, USA, 21-26 July 2017; pp. 914–922.
10. Fechter, T.; Baltas, D. One-shot learning for deformable medical image registration and periodic motion tracking. *IEEE Trans. Med Imaging* **2020**, *39*, 2506–2517.
11. Ye, M.; Kanski, M.; Yang, D.; Chang, Q.; Yan, Z.; Huang, Q.; Axel, L.; Metaxas, D. DeepTag: An Unsupervised Deep Learning Method for Motion Tracking on Cardiac Tagging Magnetic Resonance Images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual,19–25 June,2021; pp. 7261–7271.
12. Doulamis, N.; Voulodimos, A. FAST-MDL: Fast Adaptive Supervised Training of multi-layered deep learning models for consistent object tracking and classification. In Proceedings of the 2016 IEEE International Conference on Imaging Systems and Techniques (IST), Chania, Greece, 4–6 October 2016; pp. 318–323.
13. Doulamis, N. Adaptable deep learning structures for object labeling/tracking under dynamic visual environments. *Multimed. Tools Appl.* **2018**, *77*, 9651–9689.
14. Ronald, M.; Poulose, A.; Han, D.S. iSPLInception: An Inception-ResNet Deep Learning Architecture for Human Activity Recognition. *IEEE Access* **2021**, *9*, 68985–69001.
15. Boulahia, S.Y.; Amamra, A.; Madi, M.R.; Daikh, S. Early, intermediate and late fusion strategies for robust deep learning-based multimodal action recognition. *Mach. Vis. Appl.* **2021**, *32*, 1–18.

16. Zheng, C.; Wu, W.; Yang, T.; Zhu, S.; Chen, C.; Liu, R.; Shen, J.; Kehtarnavaz, N.; Shah, M. Deep learning-based human pose estimation: A survey. *arXiv* **2020**, arXiv*:2012.13392.*
17. Bin, Y.; Chen, Z.-M.; Wei, X.-S.; Chen, X.; Gao, C.; Sang, N. Structure-aware human pose estimation with graph convolutional networks. *Pattern Recognit.* **2020**, *106*, 107410.
18. Toshev, A.; Szegedy, C. Deeppose: Human Pose Estimation via Deep Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA ,23–28, June, 2014; pp. 1653–1660.
19. Chen, X.; Yuille, A.L. Articulated pose estimation by a graphical model with image dependent pairwise relations. In Proceedings of the Advances in neural information processing systems, Montreal, Quebec, Canada, 8–13 December 2014; pp. 1736–1744.
20. Hu, Q.; Yang, B.; Xie, L.; Rosa, S.; Guo, Y.; Wang, Z.; Trigoni, N.; Markham, A. Randla-net: Efficient semantic segmentation of large-scale point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11108–11117.
21. Hu, P.; Caba, F.; Wang, O.; Lin, Z.; Sclaroff, S.; Perazzi, F. Temporally distributed networks for fast video semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 8818–8827.
22. Noh, H.; Hong, S.; Han, B. Learning deconvolution network for semantic segmentation. In Proceedings of the IEEE international conference on computer vision, Santiago, Chile,7–13 December 2015; pp. 1520–1528.
23. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on computer vision and pattern recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
24. Zhu, Z.; Huang, G.; Deng, J.; Ye, Y.; Huang, J.; Chen, X.; Zhu, J.; Yang, T.; Lu, J.; Du, D. WebFace260M: A Benchmark Unveiling the Power of Million-Scale Deep Face Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual,19–25 June 2021; pp. 10492–10502.
25. Tang, J.; Su, Q.; Su, B.; Fong, S.; Cao, W.; Gong, X. Parallel ensemble learning of convolutional neural networks and local binary patterns for face recognition. *Comput. Methods Programs Biomed.* **2020**, *197*, 105622.
26. Duong, C.N.; Truong, T.-D.; Luu, K.; Quach, K.G.; Bui, H.; Roy, K. Vec2Face: Unveil Human Faces from Their Blackbox Features in Face Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 6132–6141.
27. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252.
28. Jin, A.; Yeung, S.; Jopling, J.; Krause, J.; Azagury, D.; Milstein, A.; Fei-Fei, L. Tool detection and operative skill assessment in surgical videos using region-based convolutional neural networks. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 691–699.
29. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.
30. Wang, Y.; Yao, Q.; Kwok, J.T.; Ni, L.M. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv. (CSUR)* **2020**, *53*, 1–34.
31. Shu, J.; Xu, Z.; Meng, D. Small sample learning in big data era. *arXiv* **2018**, arXiv*:1808.04572*.
32. Lu, J.; Gong, P.; Ye, J.; Zhang, C. Learning from Very Few Samples: A Survey. *arXiv* **2020**, arXiv*:2009.02653.*
33. Kingma, D.P.; Welling, M. Auto-encoding variational bayes. *arXiv* **2013**, arXiv*:1312.6114.*
34. Doersch, C. Tutorial on variational autoencoders. *arXiv* **2016**, arXiv*:1606.05908.*
35. Wei, R.; Garcia, C.; El-Sayed, A.; Peterson, V.; Mahmood, A. Variations in Variational Autoencoders-A Comparative Evaluation. *IEEE Access* **2020**, *8*, 153651–153670.
36. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the Advances in neural information processing systems, Montreal, Quebec, Canada, 8–13 December 2014; pp. 2672–2680.
37. Goodfellow, I. NIPS 2016 tutorial: Generative adversarial networks. *arXiv* **2016**, arXiv*:1701.00160.*
38. Mi, L.; Shen, M.; Zhang, J. A Probe Towards Understanding GAN and VAE Models. *arXiv* **2018**, arXiv*:1812.05676.*
39. Wei, R.; Mahmood, A. Recent Advances in Variational Autoen-coders with Representation Learning for Biomedical Informatics: A Survey. *IEEE Access* **2020**, doi:10.1109/ACCESS.2020.3048309.
40. Bengio, Y.; Courville, A.; Vincent, P. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1798–1828.
41. Schroff, F.; Kalenichenko, D.; Philbin, J. Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA ,Jun,7-12,2015; pp. 815–823.
42. Feng, Z.-H.; Hu, G.; Kittler, J.; Christmas, W.; Wu, X.-J. Cascaded collaborative regression for robust facial landmark detection trained using a mixture of synthetic and real images with dynamic weighting. IEEE Trans. Image Process. 2015, 24, 3425–3440.
43. Masi, I.; Trần, A.T.; Hassner, T.; Leksut, J.T.; Medioni, G. Do we really need to collect millions of faces for effective face recognition? In Proceedings of the European conference on Computer Vision, The Netherlands, October 11–14, 2016; pp. 579–596.
44. Suárez, J.L.; García, S.; Herrera, F. A tutorial on distance metric learning: Mathematical foundations, algorithms, experimental analysis, prospects and challenges. Neurocomputing 2021, 425, 300–322.

45. Kim, S.; Kim, D.; Cho, M.; Kwak, S. Proxy anchor loss for deep metric learning. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13-19 June, 2020; pp. 3238–3247.
46. Kulis, B. Metric learning: A survey. Foundations and trends in machine learning 2012, 5, 287-364.
47. Bellet, A.; Habrard, A.; Sebban, M. A survey on metric learning for feature vectors and structured data. arXiv 2013, arXiv:1306.6709.
48. Zhu, H.; Li, L.; Wu, J.; Dong, W.; Shi, G. MetaIQA: Deep meta-learning for no-reference image quality assessment. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13-19 June,2020; pp. 14143-14152.
49. Huisman, M.; van Rijn, J.N.; Plaat, A. A survey of deep meta-learning. Artificial Intelligence Review ,54, 4483–4541 (2021). https://doi.org/10.1007/s10462-021-10004-4
50. Vanschoren, J. Meta-learning: A survey. arXiv preprint arXiv:1810.03548 2018.
51. Tan, C.; Sun, F.; Kong, T.; Zhang, W.; Yang, C.; Liu, C. A survey on deep transfer learning. In Proceedings of the International conference on artificial neural networks, Rhodes, Greece , 4th – 7th October, 2018; pp. 270-279.
52. Weiss, K.; Khoshgoftaar, T.M.; Wang, D. A survey of transfer learning. Journal of Big data 2016, 3, 9.
53. Wang, X.; Wang, K.; Lian, S. A survey on face data augmentation. arXiv preprint arXiv:1904.11685 2019.
54. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. Journal of Big Data 2019, 6, 60.
55. Ratner, A.J.; Ehrenberg, H.; Hussain, Z.; Dunnmon, J.; Ré, C. Learning to compose domain-specific transformations for data augmentation. In Proceedings of the Advances in neural information processing systems, Long Beach, California, USA, 4-9 December,2017; pp. 3236-3246.
56. Wang, X.; Wang, K.; Lian, S. A survey on face data augmentation for the training of deep neural networks. Neural Computing and Applications 2020, 1, doi:10.1007/s00521-020-04748-3.
57. Hartig, S.M. Basic image analysis and manipulation in ImageJ. Current protocols in molecular biology 2013, 102, 14.15. 11-14.15. 12.
58. Pratt, W.K. Introduction to digital image processing; CRC press: Sep,13,2013.
59. Bartoli, A. Groupwise geometric and photometric direct image registration. IEEE Transactions on Pattern Analysis and Machine Intelligence 2008, 30, 2098-2108.
60. Holden, M. A review of geometric transformations for nonrigid body registration. IEEE transactions on medical imaging 2007, 27, 111-128.
61. Cootes, T.F.; Edwards, G.J.; Taylor, C.J. Active appearance models. IEEE Transactions on Pattern Analysis and Machine Intelligence 2001, 23, 681-685, doi:10.1109/34.927467.
62. Volker, B.; Thomas, V. A morphable model for the synthesis of 3D faces. 1999, 187-194, doi:10.1145/311535.311556.
63. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 2015.
64. Hinton, G.E.; Salakhutdinov, R.R. Reducing the dimensionality of data with neural networks. science 2006, 313, 504-507.
65. Salakhutdinov, R.; Hinton, G. Deep boltzmann machines. In Proceedings of the Artificial intelligence and statistics, Clearwater Beach, Florida USA,April,16-18,2009; pp. 448-455.
66. Grover, A.; Dhar, M.; Ermon, S. Flow-GAN: Combining Maximum Likelihood and Adversarial Learning in Genera-tive Models. In Proceedings of the Thirty-second AAAI conference on artificial intelligence, New Orleans, Louisiana, USA , February 2–7, 2017.
67. Walker, J.; Doersch, C.; Gupta, A.; Hebert, M. An uncertain future: Forecasting from static images using variational autoencoders. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands,,Oct,11-14,2016; pp. 835-851.
68. Huang, H.; Li, Z.; He, R.; Sun, Z.; Tan, T. Introvae: Introspective variational autoencoders for photographic image synthesis. arXiv preprint arXiv:1807.06358 2018.
69. Ghosh, S. Adversarial Training of Variational Auto-encoders for Continual Zero-shot Learning. arXiv e-prints 2021, arXiv: 2102.03778.
70. Ma, P.; Hu, X. A Variational Autoencoder with Deep Embedding Model for Generalized Zero-Shot Learning. In Proceedings of the Proceedings of the AAAI Conference on Artificial Intelligence, New York, New York, USA, February 7-12,2020; pp. 11733-11740.
71. Gao, R.; Hou, X.; Qin, J.; Chen, J.; Liu, L.; Zhu, F.; Zhang, Z.; Shao, L. Zero-vae-gan: Generating unseen features for generalized and transductive zero-shot learning. IEEE Transactions on Image Processing 2020, 29, 3665-3680.
72. Liu, Z.-S.; Siu, W.-C.; Chan, Y.-L. Photo-realistic image super-resolution via variational autoencoders. IEEE Transactions on Circuits and Systems for video Technology 2020, 31, 1351-1365.
73. Gatopoulos, I.; Stol, M.; Tomczak, J.M. Super-resolution variational auto-encoders. arXiv preprint arXiv:2006.05218 2020.
74. Lopez-Martin, M.; Carro, B.; Sanchez-Esguevillas, A. Variational data generative model for intrusion detection. Knowledge and Information Systems 2019, 60, 569-590.
75. Lopez-Martin, M.; Sanchez-Esguevillas, A.; Arribas, J.I.; Carro, B. Supervised contrastive learning over prototype-label embeddings for network intrusion detection. Information Fusion 2021, ISSN 1566-2535, https://doi.org/10.1016/j.inffus.2021.09.014.
76. Lopez-Martin, M.; Carro, B.; Sanchez-Esguevillas, A. IoT type-of-traffic forecasting method based on gradient boosting neural networks. *Future Generation Computer Systems* **2020**, *105*, 331-345.

77. Yi, K.; Guo, Y.; Fan, Y.; Hamann, J.; Wang, Y.G. CosmoVAE: Variational Autoencoder for CMB Image Inpainting. arXiv preprint arXiv:2001.11651 2020.
78. Tu, C.-T.; Chen, Y.-F. Facial Image Inpainting with Variational Autoencoder. In Proceedings of the 2019 2nd International Conference of Intelligent Robotic and Control Engineering (IRCE), Singapore, Singapore, Aug.25-28,2019; pp. 119-122.
79. Pihlgren, G.G.; Sandin, F.; Liwicki, M. Improving image autoencoder embeddings with perceptual loss. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19-24 July ,2020; pp. 1-7.
80. Zhou, W.; Bovik, A.C. A universal image quality index. IEEE Signal Processing Letters, Signal Processing Letters, IEEE, IEEE Signal Process. Lett. 2002, 9, 81-84, doi:10.1109/97.995823.
81. Pan, S.J.; Yang, Q. A Survey on Transfer Learning. IEEE Transactions on Knowledge and Data Engineering 2010, 22, 1345-1359, doi:10.1109/TKDE.2009.191.
82. Poulose, A.; Reddy, C.S.; Kim, J.H.; Han, D.S. Foreground Extraction Based Facial Emotion Recognition Using Deep Learning Xception Model. In Proceedings of the 2021 Twelfth International Conference on Ubiquitous and Future Networks (ICUFN), August 17 (Tue.) ~ 20 (Fri.), Jeju Island, Korea & Virtual Conference,2021; pp. 356-360.
83. Kim, J.H.; Poulose, A.; Han, D.S. The extensive usage of the facial image threshing machine for facial emotion recognition performance. Sensors 2021, 21, 2026.
84. Zheng, X.; Guo, Y.; Huang, H.; Li, Y.; He, R. A Survey of Deep Facial Attribute Analysis. International Journal of Computer Vision 2020, 128, 2002, doi:10.1007/s11263-020-01308-z.
85. Kim, H.; Mnih, A. Disentangling by factorising. In Proceedings of the International Conference on Machine Learning, Macau China February 26 - 28, 2018; pp. 2649-2658.
86. Upchurch, P.; Gardner, J.; Pleiss, G.; Pless, R.; Snavely, N.; Bala, K.; Weinberger, K. Deep Feature Interpolation for Image Content Changes. In Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA,27-30 June, 2016.
87. Masi, I.; Wu, Y.; Hassner, T.; Natarajan, P. Deep face recognition: A survey. In Proceedings of the 2018 31st SIBGRAPI conference on graphics, patterns and images (SIBGRAPI), Paraná, Brazil, Oct. 29 2018 to Nov. 1 ,2018; pp. 471-478.
88. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A. Inception-v4, inception-resnet and the impact of residual connections on learning. arXiv preprint arXiv:1602.07261 2016.
89. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, Columbus, OH, USA,23-28 June,2014; pp.1–9.
90. Hou, X.; Shen, L.; Sun, K.; Qiu, G. Deep Feature Consistent Variational Autoencoder. In Proceedings of the 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), 2017; pp. 1133–1141, doi:10.1109/WACV.2017.131.
91. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in neural information processing systems, December 3-6, Lake Tahoe, Nevada, United States,2012; pp. 1097-1105.
92. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv 2014, arXiv:1409.1556.
93. Gatys, L.A.; Ecker, A.S.; Bethge, M. A Neural Algorithm of Artistic Style. arXiv 2015, arXiv:1508.06576.
94. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of the Advances in Neural Information Processing Systems, December 5-8, Lake Tahoe, Nevada, United States,2013.
95. Sun, Y.; Wang, X.; Tang, X. Deep Learning Face Representation by Joint Identification-Verification.; The Chinese University of Hong Kong: Hong Kong, China, 2014.
96. Weinberger, K.Q.; Saul, L.K. Distance metric learning for large margin nearest neighbor classification. Journal of machine learning research 2009 Feb 1;10(2).
97. Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments., Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition, Erik Learned-Miller and Andras Ferencz and Frédéric Jurie, Oct 2008, Marseille, France. (inria-00321923)
98. Erik, L.-M.; Gary, B.H.; Aruni, R.; Haoxiang, L.; Gang, H. Labeled Faces in the Wild: A Survey. In Advances in Face Detection and Facial Image Analysis ,In Advances in face detection and facial image analysis (pp. 189-248). Springer, Cham.
99. Yi, D.; Lei, Z.; Liao, S.; Li, S.Z. Learning Face Representation from Scratch. arXiv 2014, arXiv:1411.7923.
100. Liu, Z.; Luo, P.; Wang, X.; Tang, X. Deep Learning Face Attributes in the Wild. In Proceedings of the IEEE international conference on computer vision, Las Condes, Chile, Dec 11, 2015 – Dec 18, 2015.
101. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. arXiv 2014, arXiv:1412.6980.
102. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA , 27-30 June 2016; pp. 770-778.
103. Karimi, H.; Tang, J. Decision boundary of deep neural networks: Challenges and opportunities. In Proceedings of the Proceedings of the 13th International Conference on Web Search and Data Mining, Houston TX USA, February 3 - 7,2020; pp. 919-920.
104. Liu, M.-Y.; Breuel, T.; Kautz, J. Unsupervised image-to-image translation networks. In Proceedings of the Advances in neural information processing systems, Long Beach, California, USA, 4-9 December 2017; pp. 700-708.
105. Berthelot, D.; Raffel, C.; Roy, A.; Goodfellow, I. Understanding and improving interpolation in autoencoders via an adversarial regularizer. arXiv preprint arXiv:1807.07543 2018.

106. Chen, Y.-C.; Shen, X.; Lin, Z.; Lu, X.; Pao, I.; Jia, J. Semantic component decomposition for face attribute manipulation. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15-20 June ,2019; pp. 9859-9867.
107. Ding, Z.; Xu, Y.; Xu, W.; Parmar, G.; Yang, Y.; Welling, M.; Tu, Z. Guided variational autoencoder for disentanglement learning. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA , 13-19 June,2020; pp. 7920-7929.
108. Zhu, Y.; Min, M.R.; Kadav, A.; Graf, H.P. S3VAE: Self-supervised sequential VAE for representation disentanglement and data generation. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA,Jun,13-19,2020; pp. 6538-6547.
109. Zhu, Q.; Gao, L.; Song, H.; Mao, Q. Learning to disentangle emotion factors for facial expression recognition in the wild. International Journal of Intelligent Systems 2021, 36, 2511-2527.