Generating Images of Face Poses for Pose Varying Face Recognition

Shahnas S, Sreeletha S H

Abstract-Deep learning has attracted several researchers in the field of computer vision due to its ability to perform face and object recognition tasks with high accuracy than the traditional shallow learning systems. The convolutional layers present in the deep learning systems help to successfully capture the distinctive features of the face. For biometric authentication, face recognition (FR) has been preferred due to its passive nature. Processing face images are accompanied by a series of complexities, like variation of pose, light, face expression, and make up. Although all aspects are important, the one that impacts the most face-related computer vision applications is pose. In face recognition, it has been long desired to have a method capable of bringing faces to the same pose, usually a frontal view, in order to ease recognition. Synthesizing different views of a face is still a great challenge, mostly because in non-frontal face images there are loss of information when one side of the face occludes the other. Most solutions for FR fail to perform well in cases involving extreme pose variations as in such scenarios, the convolutional layers of the deep models are unable to find discriminative parts of the face for extracting information. Most of the architectures proposed earlier deal with the scenarios where the face images used for training as well as testing the deep learning models are frontal and nearfrontal. On the contrary, here a limited number of face images at different poses is used to train the model, where a number of separate generator models learn to map a single face image at any arbitrary pose to specific poses and the discriminator performs the task of face recognition along with discriminating a synthetic face from a realworld sample. To this end, this paper proposes a representation learning by rotating the face. Here an encoder-decoder structure of the generator enables to learn a representation that is both generative and discriminative, which can be used for face image synthesis and pose-invariant face recognition. This representation is explicitly disentangled from other face variations such as pose, through the pose code provided to the decoder and pose estimation in the discriminator.

***Key Words*: pose variation, face recognition, generative adversarial network, adversarial loss**

I. INTRODUCTION

Face recognition has achieved a significant position in computer vision applications. Recently a number of face recognition techniques are developed using deep learning-based methods, due to its ability to perform face recognition tasks with high accuracy. These models fail to perform well in scenarios with pose variations in faces, since they deal with the scenarios where the face images used for training as well as testing is frontal and near-frontal.

Even though several degradations such as pose, illumination, occlusions etc deteriorates the performance of face recognition algorithms, pose variation remains to be a serious challenge in face recognition. A wide variety of approaches have been proposed to overcome the challenges in pose invariant face recognition, which can be grouped into two categories. In first category, face frontalization is applied to the input image in any arbitrary pose to synthesize a frontal-view face. This ability to generate a realistic identity preserved frontal face image helps in investigations to identify suspects. Second category is representation learning, which aims on learning the discriminative representation from the input face image. An identity representation can be obtained by modeling face frontalization or face rotation. Here the proposed framework takes best of both categories, learning an identity representation and using this representation, synthesizing faces in any arbitrary pose by face rotation.

We propose a representation learning generative adversarial network. Generative Adversarial Networks are based on the adversarial training of two CNN-based models, a generative model, which captures the true data distribution and generates images sampled from the distribution and a discriminator model, which identifies the real image from the images generated by generator. To achieve this, generator is conducted as an encoder-decoder structure. Encoder takes face image in any arbitrary pose as input and the output of the decoder is a synthetic face at a target pose, and the learnt representation bridges generator encoder and generator decoder. To predict the identity and pose of a face, discriminator is trained accordingly, while generator acts as the face rotator. Conventional GAN's uses a random noise vector as the input to generate new images, instead of that our method uses a face image f, a random noise vector n and a pose code p together as the input with an intention of generating a face image of the same identity with the target pose. Generator encoder generates a feature representation by learning the mapping from input image. Along with this feature representation, pose code and noise vector is given to the decoder for face rotation. Most of the existing face recognition techniques takes a single image as input. But our framework takes multiple images as the input. Generator encoder produces a feature representation and a coefficient for each image. All these representations are linearly combined to one single representation using the dynamically learned coefficients of each image. This single identity representation is taken along with the pose code by the decoder for face synthesis.

Revised Manuscript Received on June 15, 2020.

Shahnas S, Student, Department of Computer Science, LBSITW, Trivandrum, Kerala, India.

Sreeletha S H, Assistant Professor, Department of Computer Science, LBSITW, Trivandrum, Kerala, India.

Our generator is the essential part which is doing representation learning and face synthesis. So, two techniques that further improve generator encoderdecoder model is proposed. First one is, by replacing identity classification part of the discriminator with the generator encoder during training. And the second one is, improving the learning of decoder by regularizing the average representation of two representations from different subjects to be a valid face, assuming a convex space of face identities. Generalization ability of our model can be improved using these techniques.

II. RELATED WORKS

Zhenyao Zhu et al. [1] have described that face recognition with large pose and illumination variations is a challenging problem in computer vision. Their paper addresses this challenge by proposing a new learning-based face representation, the face identitypreserving (FIP) features. Unlike conventional face descriptors, the FIP features can significantly reduce intra-identity variances, while maintaining discriminativeness between identities.

Ying Tai and Jian Yang et al. [2] introduce the orthogonal Procrustes problem (OPP) as a model to handle pose variations existed in 2D face images. OPP seeks an optimal linear transformation between two images with different poses so as to make the transformed image best fits the other one. They integrate OPP into the regression model and propose the orthogonal Procrustes regression (OPR) model. To address the problem that the linear transformation is not suitable for handling highly non-linear pose variation, they further adopt a progressive strategy and propose the stacked OPR.

A. Face Frontalization

Generating front view of a face from a profile image is very difficult since one side of the face occludes the other. Recently some prior works are done for face frontalization, which can be classified into three categories. Statistical methods, 3D-based approaches and deep learning techniques. Hassner et al. [3] generate a frontal face from a profile image using a mean 3D face model. But still accurate 3D face reconstruction remains a challenge. In [4], a statistical model is used in which a constrained low-rank minimization problem is solved for joint frontalization and landmark localization. Kan et al. [5] propose SPAE to rotate a non-frontal face progressively to a frontal face by using auto-encoders, which is a deep learning method.

Most of the previous works frontalize only near frontal face images. In contrast, we can generate arbitrary face poses even in extreme pose variations. Adversarial loss is used by generator for improving the synthesis of face image and by discriminator in identity classification to preserve identity.

B. Representation learning

Learning a mapping from the input image to feature representation enables our model to learn a disentangled identity representation that is exclusive or invariant to pose and other variations which is essential for pose invariant face recognition. Some previous models explore face rotation and joint representation learning, where [6], [7] are relevant to our method. In [6], to untangle the identity and to view representations by processing them with different neurons and increasing the data log, multi-view perceptron is used. Yim et al. [7] discussed to rotate a face with any pose and illumination to a target pose, using a multi-task CNN and the second task is L2 lossbased reconstruction of the input.

We differ from prior works in some aspects. First, we use pose codes to explicitly disentangle the identity representation from pose variations. Second, Adversarial loss is calculated to improve the face synthesis. Finally, generalization ability of our model is improved by learning representation from multiple images.

C. Quality of face image

Image quality is an important factor in computer vision tasks. Several approaches have been proposed to measure the quality of a face image [8], [9], [10]. In [8], multiple quality factors like sharpness, brightness, contrast, illumination and focus are evaluated as a quality index of face image for face recognition. But did not consider pose variation which is a major challenge in pose varying face recognition. In this model, quality of each image is estimated in a unified GAN framework considering all factors of image quality like sharpness, brightness, illumination, focus, contrast and pose variance in the dataset. Our model generates a coefficient for each input image that indicates the quality of image.

III. PROPOSED MODEL

Our proposed model has two variations: the basic model can take one image per subject for training, termed single-image model, and the extended model can take multiple images per subject for both training and testing, termed multi-image model. First introducing the Generative Adversarial Network which is the basic network of our model, followed by describing the proposed method.

A. Generative Adversarial Network

Generative Adversarial Networks (GAN) is based on the adversarial training of two CNN-based models: (i) a generative model (G) and (ii) a discriminator model (D). The generative model captures the distribution of data and is trained in such a manner that it tries to maximize the probability of the discriminator in making a mistake. The discriminator, on the other hand, is based on a model that estimates the probability that the sample that it got is received from the training data not from the generator. The GAN's are formulated as a minimax game, where the Discriminator is trying to minimize its reward V (D, G) and the generator is trying to minimize the Discriminator's reward or in other words, maximize its loss. It can be mathematically given as:

$$\min_G \max_D \mathcal{L}_{gan} = \mathbb{E}_{\mathbf{x}\sim p_d(\mathbf{x})}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p_z(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))] \quad (1)$$

Where p_d is the distribution of real data and p_z is the distribution of generator and x and z is the sample from p_d and p_z respectively. D(x) and G(z) are the discriminator network and generator network. In the beginning of training, the samples generated from G are extremely poor and are rejected by D with high confidences. In practice, it is better for G to maximize log(D(G(z))) instead of minimizing log(1−D(G(z))). As a result, G and D are trained to alternatively optimize the following objectives:

$$\max_D \mathcal{L}_{gan}^D = \mathbb{E}_{\mathbf{x}\sim p_d(\mathbf{x})}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p_z(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))], \quad (2)$$

$$\max_G \mathcal{L}_{gan}^G = \mathbb{E}_{\mathbf{z}\sim p_z(\mathbf{z})}[\log(D(G(\mathbf{z}))]. \quad (3)$$

B. Single Image Model

Single image model has two distinctive parts. First part generates an identity representation using an encoderdecoder structure, where the representation is the output of encoder and input to the decoder for synthesizing new face images of same subject by virtually rotating the face. Second part identifies the face, not only by identity, but also considering different face variations like pose, expression, illumination. The identity representation learned by encoder includes the face variations, ie, the encoder can generate two faces of same subject in different angles. Along with the class labels, side information such as illumination and pose are employed to explicitly disentangle the variations.

I. Architecture Details

Fig 1. Represents the system architecture of single image model. Given a face image f as input with label l = (l^i, l^p) where l^i is the label of identity and l^p represents the pose code p. First, we have to learn an identity representation from the given image and then synthesizing a face image f' of the same identity l^i, but with a different pose label which is specified in the pose code p. Our discriminator D is different from conventional GAN's[12],[13],[14],[15],[16], which not only distinguish the image given is real or fake, but also do identity classification and pose classification. So, it consists of three parts, (d^r, d^i, d^p) where d^r is for the classification of real/fake image, $d^i \in R^{Ni}$ is for identity classification where Ni is the total number of subjects which is used during training and $d^p \in R^{Np}$ is for pose classification where Np is the total number of pose codes.

When a face image f is given to discriminator D, its aim is to classify that image as real or fake and to identify the person by identity classification and his pose. When we give the generated image f' = G (f, p, n) to the discriminator D , it classifies the image f' as fake image using following objectives:

$$\boldsymbol{L^D}_{\boldsymbol{gan}} = E_{f,\sim pd(f,l)}[\log d^r(f)] +$$

$$E_{\substack{f,\sim pd(f,l),\\ n\sim pz(n),p\sim pc(c)}} [\log(1 - d^r(G(f,p,n)))], \quad (4)$$

$$\boldsymbol{L_{Did}} = E_{f,\sim pd(f,l)}[\log d_{ldi}(f)], \quad (5)$$

$$\boldsymbol{L_{Dpos}} = E_{f,\sim pd(f,l)}[\log d_{lpp}(f)], \quad (6)$$

Where $d_j{}^{i}$ and $d_j{}^{p}$ is the jth element of d^i and d^p respectively. For training of D, the weighted average of all objectives was taken, ie,

$$\boldsymbol{maxL_D} = \lambda_g L_{Dgan} + \lambda_i L_{Did} + \lambda_p L_{Dpos} \quad (7)$$

Where $\lambda_g = \lambda_i = \lambda_p = 1$

Considering generator, it consists of an encoder, G_e and a decoder, G_d. Generator encoder learns an identity representation, G(f)=G_e(f) from the given face image f. Generator decoder generates a face image f ≐ G_d (Ge(f), p, n) using this representation, pose code p and random noise vector n with an identity l^i. Pose code which is represented by p ϵ R ^{Np}, is a one-hot vector where the target pose l^t equals 1.Generator,G tries to fool discriminator, D to classify the generated image f to the identity of the input image f and the' target pose by following equations:

$$\boldsymbol{L^G}_{\boldsymbol{gan}} = [\log d^r(G(f,p,n))], \quad (8)$$

$$\boldsymbol{L^G}_{\boldsymbol{id}} = [\log d_l{}^{d}{}_{i}(G(f,p,n))], \quad (9)$$

$$\boldsymbol{L^G}_{\boldsymbol{pos}} = [\log d_l{}^{p}{}_{t}(G(f,p,n))], \quad (10)$$

To train the generator G, weighted average of each objectives is used.ie,

$$\boldsymbol{maxL_G} = \mu_g L_{Ggan} + \mu_i L_{Gid} + \mu_p L_{Gpos} \quad (11)$$

Where $\mu_g = \mu_i = \mu_p = 1$

Generator, G and Discriminator, D improves each other during training by D being stronger in identifying synthetic images and classifying different poses. While generator synthesize identity preserved face images with the target pose to compete with the discriminator, D. Training is illustrated in Fig 2.

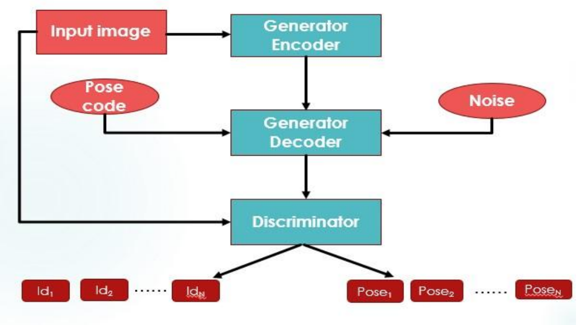

Fig 1. System architecture of single image model

II. Multitask CNN Structure

Proposed convolutional neural network adopts CASIA-WebFace [11] with batch normalization for generator encoder G_e and discriminator D.

Network includes 10 convolution layer, 5 pooling layers where the first four layers uses max operator and fifth layer takes the average and one fully connected layer. The network structure is given in Table 1 and Table 2.

Generator encoder G_e and generator decoder G_d are bridged using the identity representation G(f) ϵ R^{Nf},which is the Avgpool output of the G_e. G(f) is concatenated with the noise vector n and pose code p. A number of fractionally-strided convolutional layers (FConv) are used to transform the $(Nf+ \ Np+ \ Nd)$ dimensional vector into a generated image f' = G (f, p, n) which is same as f. Size of all the filters in the network is 3 × 3.

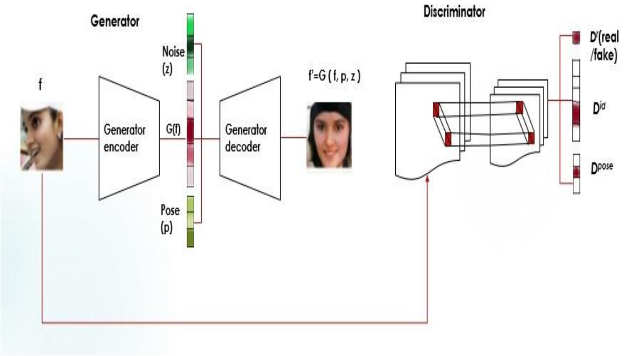

Fig 2. Training part of the model

Prior works [12] requires face image pairs at different poses of the same subject to be included in the training data, which is hard to achieve in wild datasets. Our model does not require such image pairs so it can be used in wild datasets.

Table 1. Multitask CNN structure of Generator encoder G_e and Discriminator D

G_e and D			
Layer	**Type**	**Filter/Stride**	**Output**
Conv11 Conv12	Convolution Convolution	3 × 3/1 3 × 3/1	96 × 96 × 32 96 × 96 × 64
Pool1	Maxpooling	2 × 2/2	48 × 48 × 64
Conv21 Conv22	Convolution Convolution	3 × 3/1 3 × 3/1	48 × 48 × 64 48 × 48 × 128
Pool2	Maxpooling	2 × 2/2	24 × 24 × 128
Conv31 Conv32	Convolution Convolution	3 × 3/1 3 × 3/1	24 × 24 × 96 24 × 24 × 192
Pool3	Maxpooling	2 × 2/2	12 × 12 × 192
Conv41 Conv42	Convolution Convolution	3 × 3/1 3 × 3/1	12 × 12 × 128 12 × 12 × 256
Pool4	Maxpooling	2 × 2/2	6 × 6 × 256
Conv51 Conv52	Convolution Convolution	3 × 3/1 3 × 3/1	6 × 6 × 160 6 × 6 × $(Nf + 1)$
Pool5	Avgpooling	6 × 6/1	1 × 1 × $(Nf + 1)$
FC (only D)	Fully connected		Ni+Np+1

To initialize training, a training image is given and for each pose view, pose codes are randomly sampled with equal probability. To assign multiple pose codes to a single training image, random sampling is done at each epoch during training. Randomly sampling each dimension from a uniform distribution in the range [1,1] makes the noise vector.

III. Multi-Image Multi pose Model

Our single image model extracts an identity representation from a single image and generates an image of the same subject in a specific pose. In multiimage model, multiple images per subject is used for training such that it can learn a better identity representation than the single image model. Template -to- template matching is used for testing the image.

Table 2. Multitask CNN structure of Generator decoder G_d an image of the same subject in a specific pose.

G_d			
Layer	**Type**	**Filter/Stride**	**Output**
FC FConv52 FConv51	Fully connected Fully connected	3 × 3/1 3 × 3/1	6 × 6 × 320 6 × 6 × 160 6 × 6 × 256
Upsample5	Upsampling	3 × 3/2	12 × 12 × 256
FConv42 FConv41	Fully connected Fully connected	3 × 3/1 3 × 3/1	12 × 12 × 128 12 × 12 × 192
Upsample4	Upsampling	3 × 3/2	24 × 24 × 192
FConv32 FConv31	Fully connected Fully connected	3 × 3/1 3 × 3/1	24 × 24 × 96 24 × 24 × 192
Upsample3	Upsampling	3 × 3/2	48 × 48 × 128
FConv22 FConv21	Fully connected Fully connected	3 × 3/1 3 × 3/1	48 × 48 × 64 48 × 48 × 64
Upsample2	Upsampling	3 × 3/2	96 × 96 × 64
FConv12 FConv11	Fully connected Fully connected	3 × 3/1 3 × 3/1	96 × 96 × 32 96 × 96 × 3

Fig 3. illustrates multi-image model which consists of discriminator same as the single image model and a generator different from single image model, that it consists of n number of encoder module where n is the number of input images of same identity with different poses. Along with the feature representation, encoder calculates a coefficient βi for each image which gives better quality for the representation.

Weighted average of the coefficients gives the fused representation of the n input images as follows

$$(f1, f2, \dots fn) = \frac{\sum_{i=1}^{n} \beta i G(fi)}{\sum_{i=1}^{n} \beta i} \tag{12}$$

The fused identity representation is combined with pose code p and noise z and given into the generator decoder to generate new image, which is expected to have same identity as all the input images and to the target pose specified in the pose code. Hence generator in multi-modal learns by (n+1) terms.ie,

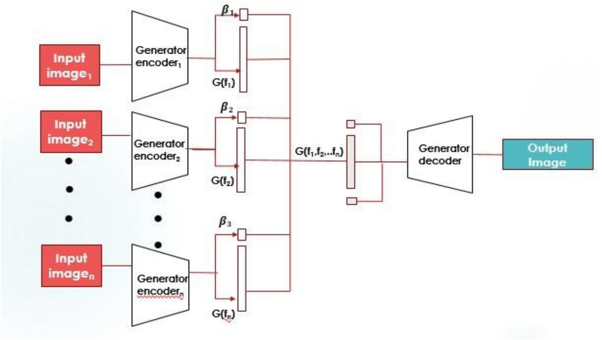

Fig 3. Multi-image Multi-pose model

$$L^{G}{}_{gan} = \sum_{i=1}^{n} E\left[\log d^{r}\left(G_{(fi, p, z))}\right] + \right.$$

$$[\log d^{r}(G(f1, f2, \dots, fn, p, z))] \tag{13}$$

$$L^{G}{}_{id} = \sum_{i=1}^{n} E\left[\log d_{l^{i}}^{d}\left(G_{(fi, p, z))}\right] + \right.$$

$$E\left[\log d^{d}{}_{i}\left(G(f1, f2_{l}, \dots, fn, p, z)\right)\right] \tag{14}$$

$$L^{G}{}_{pos} = \sum_{i=1}^{n} E\left[\log d_{l^{t}}^{p}\left(G_{(fi, p, z))}\right] + \right.$$

$$[\log d_{l}{}^{p}{}_{t}(G(f1, f2, \dots, fn, p, z))] \tag{15}$$

Network structure is modified by adding one more convolutional filter to the layer before AvgPool to calculate coefficient β, specifically at the end of generator encoder. Sigmoid activation function is used to constrain β in the range of [0, 1]. Mainly, multiimage model can be used for better quality face synthesis.

IV. EXPERIMENTAL SETTINGS & DATASETS

Our model can work in both wild datasets and controlled datasets. For implementation, first we have to align all faces to a canonical view of size 110 × 110 and for data augmentation, randomly sample a region of size 96 × 96 from the aligned image.

Implementations are coded in python-keras platform with tensorflow backend in google colab. Image intensity is linearly scaled in the range [-1, 1].3D face alignment [17],[18] is applied to classify each image in one of specific pose to provide pose labels. Batch size is kept to 64. Weights were randomly initialized with a standard deviation 0.2 and trained on GPU environment for 2-3 hours.

CFP (Celebrity in Frontal Profile) dataset is used for testing. Database includes 500 individuals each having 10 frontal and 4 profile images. So, in total it consists of 5000 frontal faces and 2000 profile faces. Frontalfrontal (FF) and frontal-profile (FP) face verification is included in evaluation protocol, each having 10 folders with 350 different-person pairs and 350 sameperson pairs.

V. CONCLUSIONS

This paper learns an identity representation for pose varying face recognition. For the representation learning, GAN network is used. A generator with encoder-decoder network is constructed. Generator encoder is used for learning features from the input image and along with this feature representation, a pose code and noise are fed into the generator decoder. Noise is for considering the image background, illuminations and so on. Using the data given, decoder generates a new image with the same identity as the input image but with a different pose as specified in the pose code. Discriminator in GAN network is used for pose classification and face recognition. Also, this paper proposes a multi-image multi-pose model to improve the quality of generating image. From multiple image of same identity given as input, a confident coefficient is estimated from each image and by taking the weighted average of coefficients, a better identity representation can be used for decoding.

REFERENCES

1. Zhenyao Zhu, Ping Luo, Xiaogang Wang Xiaoou Tang, "Deep Learning Identity-Preserving Face Space" 2013 IEEE International Conference on Computer Vision.
2. Ying Tai, Jian Yang, Yigong Zhang, Lei Luo, Jianjun Qian, and Yu Chen, "Face Recognition with Pose Variations and Misalignment via Orthogonal Procrustes Regression" IEEE
3. Transactions on Image Processing, Vol. 25, No. 6, June 2016
4. T. Hassner, S. Harel, E. Paz, and R. Enbar,
5. "Effective face frontalization in unconstrained images," in CVPR, 2015.
6. C. Sagonas, Y. Panagakis, S. Zafeiriou, and M. Pantic, "Robust statistical face frontalization," in ICCV, 2015.
7. M. Kan, S. Shan, H. Chang, and X. Chen,
8. "Stacked Progressive AutoEncoders (SPAE) for face recognition across poses," in CVPR, 2014. [6] Z. Zhu, P. Luo, X. Wang, and X. Tang, "Multiview perceptron: a deep model for learning face identity and view representations,"in NIPS,2014. [7] J. Yim, H. Jung, B. Yoo, C. Choi, D. Park, and J.
9. Kim, "Rotating your face using multi-task deep neural network," in CVPR, 2015.
10. A. Abaza, M. A. Harrison, T. Bourlai, and A. Ross, "Design and evaluation of photometric image quality measures for effective face recognition," IET Biometrics, 2014.
11. M.Abdel-Mottaleband M.H. Mahoor,
12. "Application notes-algorithms for assessing the quality of facial images," IEEE Computational Intelligence Magazine, 2007. [10] N. Ozay, Y. Tong, F. Wheeler, and X. Liu, "Improving face recognition with a quality-based probabilistic framework," in CVPRW, 2009.
13. D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," arXiv:1411.7923, 2014.
14. L. Tran, X. Yin, and X. Liu, "Disentangled Representation Learning GAN for pose-invariant face recognition," in CVPR, 2017. [13] M. Mirza and S. Osindero, "Conditional generative adversarial nets," arXiv:1411.1784, 2014.
15. H. Kwak and B.-T. Zhang, "Ways of conditioning generative adversarial networks," in NIPSW, 2016.
16. A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in ICML, 2017.
17. A. Makhzani, J. Shlens, N. Jaitly, and I. Goodfellow, "Adversarial autoencoders," in ICLRW, 2015.

Retrieval Number: F9998038620/2020©BEIESP
DOI:10.35940/ijrte.F9998.079220

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

18. A. Jourabloo, X. Liu, M. Ye, and L. Ren, “Poseinvariant face alignment with a single CNN,” in ICCV, 2017.
19. A. Jourabloo and X. Liu, “Pose-invariant face alignment via CNN-based dense 3D model fitting,” IJCV, 2017. [67] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in ICLR, 2015.