May – June 2020
ISSN: 0193-4120 Page No. 7836 - 7840

Surveillance based on Representation learning using Generative Adversarial Networks

Nibi Maouriyan[1], S Shanthi[2] , Aravinth Krishna KN[3] , Mukilan E [4]
[1,3,4] Student , Department of Computer Science and Engineering, SRM Valliammai Engineering College
[2] Assistant Professor, Department of Computer Science and Engineering, SRM Valliammai Engineering College

Article Info
Volume 83
Page Number:7836 - 7840
Publication Issue:
May-June 2020

***Abstract*:**
Face recognition is one of the most advancing topics in Deep Learning due to its wide applications/ Recently, great progress has been made in computer vision for security and biometric applications. However that it does not offer a high degree of pose variation—The major challenge in face recognition is the variance of face attributes. The face recognition also fails when the Face is covered with arbitrary masks. To overcome these challenges,we propose a surveillance system using Disentangled Representation learning- Generative Adversarial Network (DR-GAN) a encoder-decoder structure implementing Wasserstein loss, which can be grouped into two categories. First, we apply face implanting which reconstructs an image which is masked or noisy. Second, some work applies*face frontalization* on the input image to generate pose-invariant faces, where traditional face-detection algorithms are applicable, or an identity representation can be obtained using the face rotation.Together they can be used to search/detect a human face with only masked or profile image as input.We use a self collected dataset of Indian Faces to improve accuracy on Indian Faces.

Article History
Article Received: *19 November 2019*
Revised: *27 January 2020*
Accepted: *24 February 2020*
Publication: *18 May 2020*

Keywords: Representation learning, generative adversarial network, face inplanting, surveillance pose-invariant face recognition, face rotation and frontalization.

I.INTRODUCTION

The present surveillance systems are only used to monitor the crime , during the incident of the crime a criminal's profile image or only a part of the image can be recovered.It is very difficult to 1) Identify the person in existing database based on the features recovered and 2) Detect the query person in new surveillance videos.

Humans can understand facial structures and create an imaginative map on real images from a masked image or profile image based on visible features.Humans can draw and visualize their understanding. However it is very difficult for the machine to understand the facial structures and patterns of a person.The state-of-the-art techniques try to complete this task.
Autoencoder[2] systems use the environment pixels to reconstruct the images.In this paper we show the possibility of learning and predicting the human faces using convolutional neural networks (CNNs).Different models that have shown success in the task of understanding.

With an image with a masked region we train a generative adversarial network to fill the missing area and pixel values. The model is called context encoding since it understands the context of the image and makes a disentangled representation.
Disentangled representation [1] is a process of learning only parts of the features separately to learn the structure , for example a normal CNN will learn all the features of a human and plot it in a multi-dimensional graph and based on the graph location it predicts whether the human is an infant or adult.In disentangled representation it can learn only a feature like nose width and compare it to the

threshold of the nose width of infant and adult and make a classification.

Like autoencoders, GAN is trained using an encoder and decoder mechanism in a completely unsupervised manner.For the success of this task our model needs to understand the content of the image and produce acceptable results for missing regions[3].

The GAN model must also make face frontalization converting the profile image from 1) various degrees of angles , 2) Illumination 3) Noise to a frontal image.

We evaluate the generator(encoder) and discriminator(decoder) separately , each having a loss function and the generated fake images are evaluated and the feedback is backward propagated to the encoder - decoder structure[4].To increase the accuracy in Indian surveillance system , we collected and pre-processed the Indian face Image dataset of various frontal and profile images of each individual.

II.RELATED WORKS

The tasks such as object detection, classification and segmentation on semantic images have made tremendous progress using computer vision. . The major advancement is due to Convolutional Neural Networks(CNN).It has lead to solve larger problems such as unsupervised learning and generation of natural images.Generalisation of classification tasks with over a million labeled data works well on ImageNet trained using CNN. Autoencoders[5] are some of the earliest work on unsupervised learning. [3, 20]. Denoising encoders are used to resconstruct images from corrupted images.Denoising autoencoders are a variant of context encoders it requires more semantic information to undo since the noise added to the input is much larger. Weakly-supervised and self-supervised learning Very recently, there has been significant interest in learning meaningful representations using weakly-supervised and self supervised learning. Non-visual sensors to read Ego-Motion are used as supervised signals for training features. Visual Memex is a technique for using context to non-parametrical model object relations and to predict missing regions in scenes, while others used context to create correlation for unsupervised object discovery. However,both approaches does not perform the representation learning andusesmanually selected features. features and did not perform any representation learning.

Recently, a way to train the unsupervised deep feature representation within an image of used the task of predicting the relative positions of nearest regions.
We share the same high-level goals.

III. GENERATIVE ADVERSARIAL NETWORKS

Generative Adversarial Network(GAN)[6] was invented by Ian Goodfellow and his colleagues in 2014.It is a class of machine learning systems. Two neural networks such as Generator and discriminator contest with each other in a game. From the given training dataset, this method learns to generate new data with the same features as the training dataset. For example, photographs trained using GAN can generate new photographs that will look like a real photo, having many realistic characteristics. GAN is form of generative model for unsupervised learning,semi-supervised learning, fully supervised learning, and reinforcement learning. The generative network generates images based on the given training dataset while the discriminative network evaluates them as fake or real. This process works based on data distribution . Typically, the generative network learns to produce images from the training dataset , while the discriminative network distinguishes those images from normal distribution.The aim of generator is to fool discriminator by producing new images and to make the discriminator to believe that the images are real. Discriminator has the known dataset as the initial training data and it is trained by presenting it with samples from the training dataset, until it achieves acceptable accuracy. The generator is trained based on the result of discriminator(Whether it has succeded in fooling the discriminator or not). Typically the generator's input is random that is sampled from a predefined latent space. Thereafter, ouput produced by the generator are evaluated by the discriminator. Backpropagation is applied to both generator and discriminator so that generator can produce better images, while the discriminator becomes more skilled in identifying fake images.

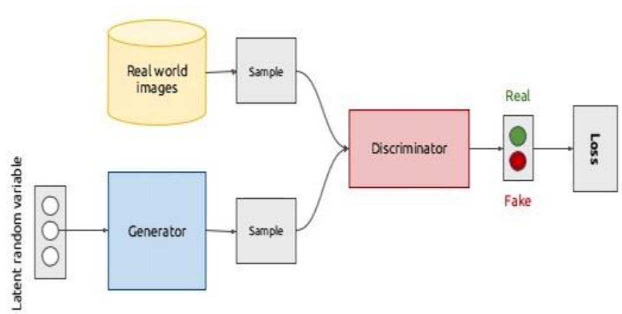

Figure 1 : GAN ARCHITECTURE[7]

TEST Engineering & Management

May – June 2020
ISSN: 0193-4120 Page No. 7836 - 7840

IV. DATASET

We collected the dataset of 50 Indian faces. The mean age of the people in the dataset is 20 years . It contains 20% Females. The dataset is collected for two purposes.

1) For face inpainting , images of 50 individuals containing five images each : three Frontal and two Profile.

2) For face frontalization . images of 14 individuals containing twenty images each : ten Frontal and ten Profile.

Pre-processing has been applied to resize the images into 128x128x3 RGB pictures of JPG format.

For face inpainting technique the images are patched with a square mask at bottom using logical AND with images and mask in opencv.Two seperate directories are created for original and patched (corrupted) for training.For face frontalization the images of size 128x128 are directly used.

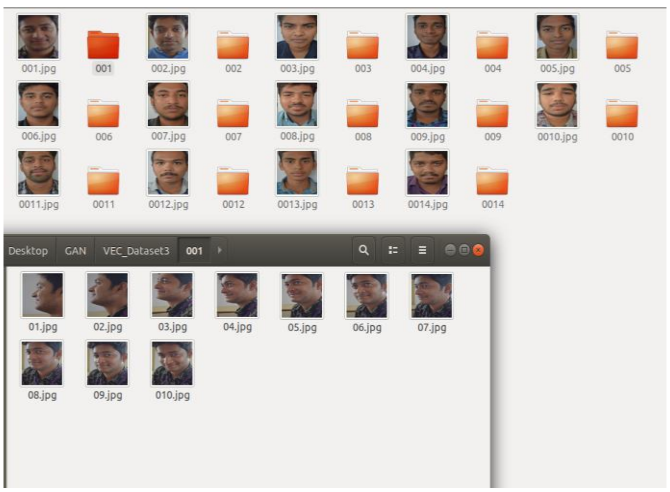

Figure 2: WGAN Algorithm

V.LOSS FUNCTION

The Wasserstein GAN or WGAN, was introduced by Martin Arjovsky, et al. in their 2017 paper titled "Wasserstein GAN." This method is an attempt to train the generator model in an alternative way to improve the approximation of distribution of data found in the given training dataset. The discriminator is replaced by the WGAN with a critic model which gives scores to predict the image as real or fake.

This change is because of a mathematical argument that training the generator should seek a minimization of the distance between the distribution of the data observed in the training dataset and the distribution observed in generated examples. The different distribution distance measurements, such as KL divergence, JS divergence, and the Earth-Mover (EM) distance, are contrasted to Wasserstein.

Implementation Details of the Wasserstein GAN

The WGAN[8] implementation requires a few minor changes to the standard deep convolutional GAN, or DCGAN although the grounding theory is very dense. Those changes are as follows:

- Instead of using sigmoid a linear activation function can be used in the output layer of the critic model

- To train the critic and generator model Wasserstein loss can be used which promote larger difference between the scores for real and the generated images.

- Critic model weights are constrained to a range after each batch update ([-0.01,0.01]).

Algorithm 1 WGAN, our proposed algorithm. All experiments in the paper used the default values $\alpha = 0.00005$, $c = 0.01$, $m = 64$, $n_{\text{critic}} = 5$.

Require: : α, the learning rate. c, the clipping parameter. m, the batch size. n_{critic}, the number of iterations of the critic per generator iteration.
Require: : w_0, initial critic parameters. θ_0, initial generator's parameters.

while θ has not converged **do**
  for $t = 0, ..., n_{\text{critic}}$ **do**
    Sample $\{x^{(i)}\}_{i=1}^m \sim \mathbb{P}_r$ a batch from the real data.
    Sample $\{z^{(i)}\}_{i=1}^m \sim p(z)$ a batch of prior samples.
    $g_w \leftarrow \nabla_w \left[\frac{1}{m}\sum_{i=1}^m f_w(x^{(i)}) - \frac{1}{m}\sum_{i=1}^m f_w(g_\theta(z^{(i)}))\right]$
    $w \leftarrow w + \alpha \cdot \text{RMSProp}(w, g_w)$
    $w \leftarrow \text{clip}(w, -c, c)$
  end for
  Sample $\{z^{(i)}\}_{i=1}^m \sim p(z)$ a batch of prior samples.
  $g_\theta \leftarrow -\nabla_\theta \frac{1}{m}\sum_{i=1}^m f_w(g_\theta(z^{(i)}))$
  $\theta \leftarrow \theta - \alpha \cdot \text{RMSProp}(\theta, g_\theta)$
end while

` Figure 3: WGAN Algorithm

The above given algorithm is quite similar to the original GAN. Inorder to implement WGAN, we should notice few things from the above:

1.No log in the loss. The output of D is no longer a probability, hence we do not apply sigmoid at the output ofD
2.Clip the weight of D
3.D should be trained more than G

May – June 2020
ISSN: 0193-4120 Page No. 7836 - 7840

4.Use RMSProp instead of ADAM
5.Lower learning rate, the paper uses $\alpha = 0.00005$

VI.IMPLEMENTATION DETAILS

The recently proposed approach such as stochastic gradient descent solver, RMS prop for optimization purpose. The constant mean value is used to fill the missing regions in the masked input image.

The experiment of replacing all pooling layers with convolutions of the same kernel size and stride results in finer inpainting but the overall stride of the network remains the same. There is no specific reason to use pooling for reconstruction based networks.

Spacial invariance which is provided by pooling may cause harm for reconstruction-based training. Orginal AlexNet architecture (with pooling) is used for all feature learning results.

VII.RESULTS

A) INPAINTING

Figure 4: Ground Truth

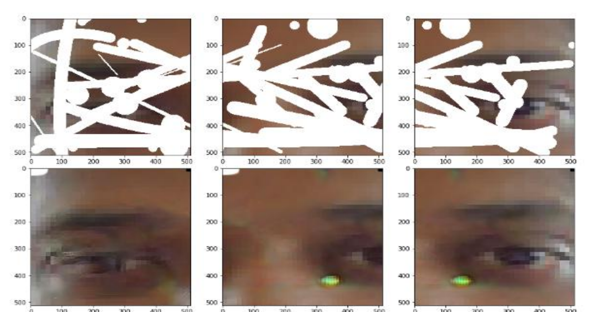

Figure 5(a):Inpainting Regions A

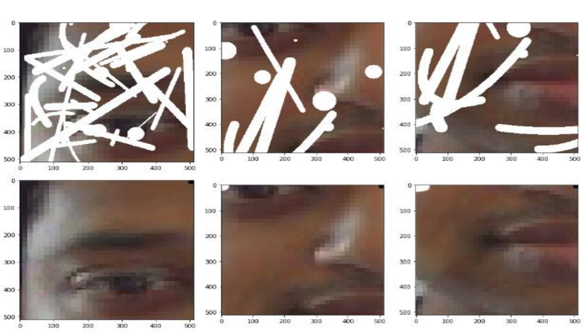

Figure 5(b):Inpainting Regions B

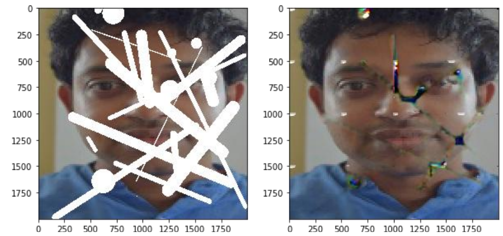

Figure 6: Splitting the image 512x512

B) Rotation

Figure 7:Rotation Trained with CelebA and Our Dataset

VIII.CONCLUSIONS

The face inpainting provides promising results when it is split into 512x512 small bits , however when the whole image is provided as a single 512x512 image it is not able impaint with much detailing.

For face rotation using a big dataset of images belonging to over 400 individuals our model is able to generate a face using the learned structures however it does a lot of generalisation. While using only our dataset the images are not enough for perfect detailing.

Hence the dataset we collected is enough for face inpainting but it would require a larger dataset to generate frontalised Indian Faces.

IX.REFERENCES

1. L. Tran, X. Yin and X. Liu, "Representation Learning by Rotating Your Faces," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 12, pp. 3007-3021, 1 Dec. 2019.doi: 10.1109/TPAMI.2018.2868350
2. Pathak, Deepak et al. "Context Encoders: Feature Learning by Inpainting." *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016): 2536-2544.
3. P. Agrawal, J. Carreira, and J. Malik. Learning to see by moving. ICCV, 2015.

4. Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2014, pp. 1701–1708.
5. D. P. Kingma and M. Welling. Auto-encoding variational bayes. ICLR, 2014.
6. Goodfellow, J. Pouget-Abadie , M. Mirza, B. Xu, D. Warde-Farley,S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets,"in Proc. Int. Conf. Neural Inf. Process. Syst., 2014, pp. 2672–2680.
7. Y. Bengio. Learning deep architectures for ai. Foundations and trends in Machine Learning, 2009.
8. Arjovsky, Martín et al. "Wasserstein GAN." *ArXiv* abs/1701.07875 (2017): n. pag.